

NASA Contractor Report 145346

REQUIREMENTS FOR KALMAN FILTERING ON THE GE-701
WHOLE WORD COMPUTER

S. Pines and S. F. Schmidt

(NASA-CR-145346) REQUIREMENTS FOR KALMAN
FILTERING ON THE GE-701 WHOLE WORD COMPUTER
(Analytical Mechanics Associates, Inc.)
97 p HC A05/MF A01

CSCL 09B

N78-23752

G3/60 16757
Unclas

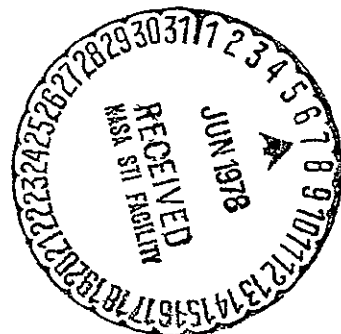
ANALYTICAL MECHANICS ASSOCIATES, INC.
90 Jericho Turnpike
Jericho, NY 11753

NASA Contract NAS1-14311
April 1978



National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665



REQUIREMENTS FOR KALMAN FILTERING ON THE GE-701 WHOLE WORD COMPUTER

S. Pines
S. F. Schmidt

Analytical Mechanics Associates, Inc.

SUMMARY

This report contains the results of a study to determine scaling, storage, and word length requirements for programming the Kalman filter on the GE-701 Whole Word Computer. Simulation tests are presented which indicate that the Kalman filter, using a square root formulation with process noise added, utilizing MLS, radar altimeters, and airspeed as navigation aids, may be programmed for the GE-701 computer to successfully navigate and control the Boeing B737-100 during landing approach, landing rollout, and turnoff.

The report contains flow charts, equations, computer storage, scaling, and word length recommendations for the Kalman filter on the GE-701 Whole Word computer.

INTRODUCTION

Kalman filtering has been shown to provide excellent navigation with smooth transition from the non-critical flight phases using TACAN to critical flight phases of capture, flare, and decrab using the Microwave Landing System (MLS) and radar altimeters. As part of the Terminal Configured Vehical program (TCV), jointly sponsored by NASA Langley Research Center and FAA, this study was undertaken to 1) investigate the computer requirements to implement the Kalman filter navi-

gation equations for the GE-701 Whole Word computer on the B737-100 aircraft, and 2) demonstrate by simulation the possibility of supplying guidance information for control of the aircraft during landing, rollout, and turnoff.

For the purpose of this study, a simulation of the GE-701 computer was incorporated into the existing ALERT program described in Ref. 1. The word length and the available restricted arithmetic capabilities of the GE-701 computer, as described in Ref. 2, were simulated. The program was designed to artificially extend the GE-701 computer word length from 16 bits (including sign) to 60 bits (including sign). By running the simulation at various bit-size levels, the study proved that a 16-bit machine could provide a stable Kalman navigation filter for guidance and control provided that additions and subtractions could be carried out in double-precision (32 bits) add and subtract instructions, and that square root covariance techniques and process noise addition were executed.

The Kalman filter is known to require extensive computations because of the need to maintain a positive definite reasonable approximation to the true covariance matrix of the solved-for variables. In order not to occupy the central computer with excessive computations, the recommended logic for a successful Kalman filter requires a three-tier operational capability. The first spews out the integrated variables necessary for guidance and displays. These must usually be available at a rate of 20 Hz. The second tier computes the navigational aid measurements, the residuals, and the partial derivatives. These are required at a much slower rate, usually at 5 Hz. Finally, the heavy computations for updating the corrections to the solved-for variables and propagating the covariance matrix are reserved for the slowest rate, say every 2 seconds or 0.5 Hz. In this manner, the central computer can be kept relatively free for other functions most of the time.

$$\begin{array}{r} 1806 \\ 422 \\ \hline 2233 \end{array}$$

TABLE OF CONTENTS

<u>Section</u>	<u>Item</u>	<u>Page</u>
	SUMMARY	i
	INTRODUCTION	i
I	SIMULATION OF THE GE-701 WHOLE WORD COMPUTER	1
II	FLOW CHARTS AND EXECUTIVE ROUTINES FOR THE KALMAN FILTER	2
III	EQUATIONS FOR FOREGROUND COMPUTATIONS	9
IV	EQUATIONS FOR MIDGROUND COMPUTATIONS	14
V	EQUATIONS FOR BACKGROUND COMPUTATIONS	25
VI	INITIALIZATION	32
VII	STORAGE AND SCALING	37
VIII	COMPARISON OF KALMAN FILTER NAVIGATION USING 16 BITS VERSUS 60 BITS	87
IX	REFERENCES	93

LIST OF FIGURES

<u>Figure Number</u>	<u>Title</u>	<u>Page</u>
I	Macro Flow Chart of Executive for Kalman Filter	3
II	Macro Flow Chart of the 20 Hz Foreground Calculations for the MCP-701 Computer	5
III	Macro Flow Chart of the 5 Hz Midground Calculations for the MCP-701 Computer	7
IV	Macro Flow Chart of the .5 Hz Background Calculations for the MCP-701 Computer	8
Va	16-Bit Simulation of the B-737 Kalman Filter	88
Vb	16-Bit Simulation of the B-737 Kalman Filter	89
VIa	60-Bit Simulation of the B-737 Kalman Filter	90
VIb	60-Bit Simulation of the B-737 Kalman Filter	91

LIST OF TABLES

<u>Table Number</u>	<u>Title</u>	<u>Page</u>
1	Foreground	38
2	Midground	42
3	Background	52

I. SIMULATION OF THE GE-701 WHOLE WORD COMPUTER

The GE-701 computer is described in Ref. 2. The computer is a fixed-point machine with a normal word length of 16 bits. Double-precision (32-bit) addition and subtraction are readily available; however, double-precision multiplication and division are available only in software code and its use is to be avoided, if at all possible.

The ground rule employed throughout the simulation was to permit double-precision addition and subtraction, but to restrict all products and divides to the single-precision mode. Since the ALERT program, which contains the simulation of the aircraft, is a floating-point program, the simulation of the GE computer with its restricted word length was carried out in a floating-point mode. Every FORTRAN instruction was decomposed into a sequence of add, subtract, multiply, and divide instructions. Each variable entering the operation was masked to a 16-bit word including sign. The result of each product or divide was again masked to a 16-bit word including sign and, if the resulting variable was a sum, the sum was accumulated as a double-precision word of 32 bits including sign. The final value of each variable was a 16-bit word.

While a fixed-point machine cannot hope to achieve the accuracy of a floating-point machine of the same word length, it can nevertheless come reasonably close to the same accuracy provided the proper scaling is available for the variables entering each arithmetic operation. For this purpose, the simulation program kept track of the algebraically smallest and largest value attained by each variable, so that the range of each variable over the entire trajectory is available for guidance in choosing the proper scale factor. This information is available in this report for each variable and every intermediate scratch variable.

II. FLOW CHARTS AND EXECUTIVE ROUTINES FOR THE KALMAN FILTER

Figure I is a macro flow chart for the executive driver for the three-tier navigation system for the Kalman filter. The executive is arranged in a manner that permits calculations at cycling frequencies of 20 Hz, 5 Hz, and .5 Hz.

At power-up the system is initialized, check-sum calculations for validating instructions and constants for computer malfunctions are executed. When the 20 Hz-interrupt occurs, the executive transfers to the top of Figure I. The first operation is to save the A and Q registers and the program locations where the interrupt occurred in the software equivalent of a push-down stack. The push-down stack is needed since several interrupts can occur before all the restores are executed. The stack is so arranged that complete recovery from the interrupt is made by restoring the registers to the last value before the interrupt and branching to this last location when interrupted.

Following completion of the foreground calculations, the foreground overflow flag, FGOV, is set zero and the test made to determine if it is time to execute the 5 Hz midground calculations. If not, the registers are restored from the push-down stack and a branch to the interrupt location is made. Note that the location branched to has the order of priority 1) do the 20Hz calculations if they are not completed; 2) do the 5 Hz calculations if they are not completed; or 3) do the far background wait loop. If it is time to start the 5. Hz midground calculations, the 5Hz midground overflow flag, MDOV, is tested to determine whether a malfunction has occurred in the 5 Hz logic. If not, MDOV is set unity and the 5 Hz calculations are initiated. Following completion, MDOV is set zero and the test made to determine if it is time to start the .5 Hz calculations. If not, the registers are restored and a branch to the interrupt location made. In this instance, the branch is according to the priority 1) do the 5 Hz logic if the calculations have not been completed, or 2) do the wait loop in far background.

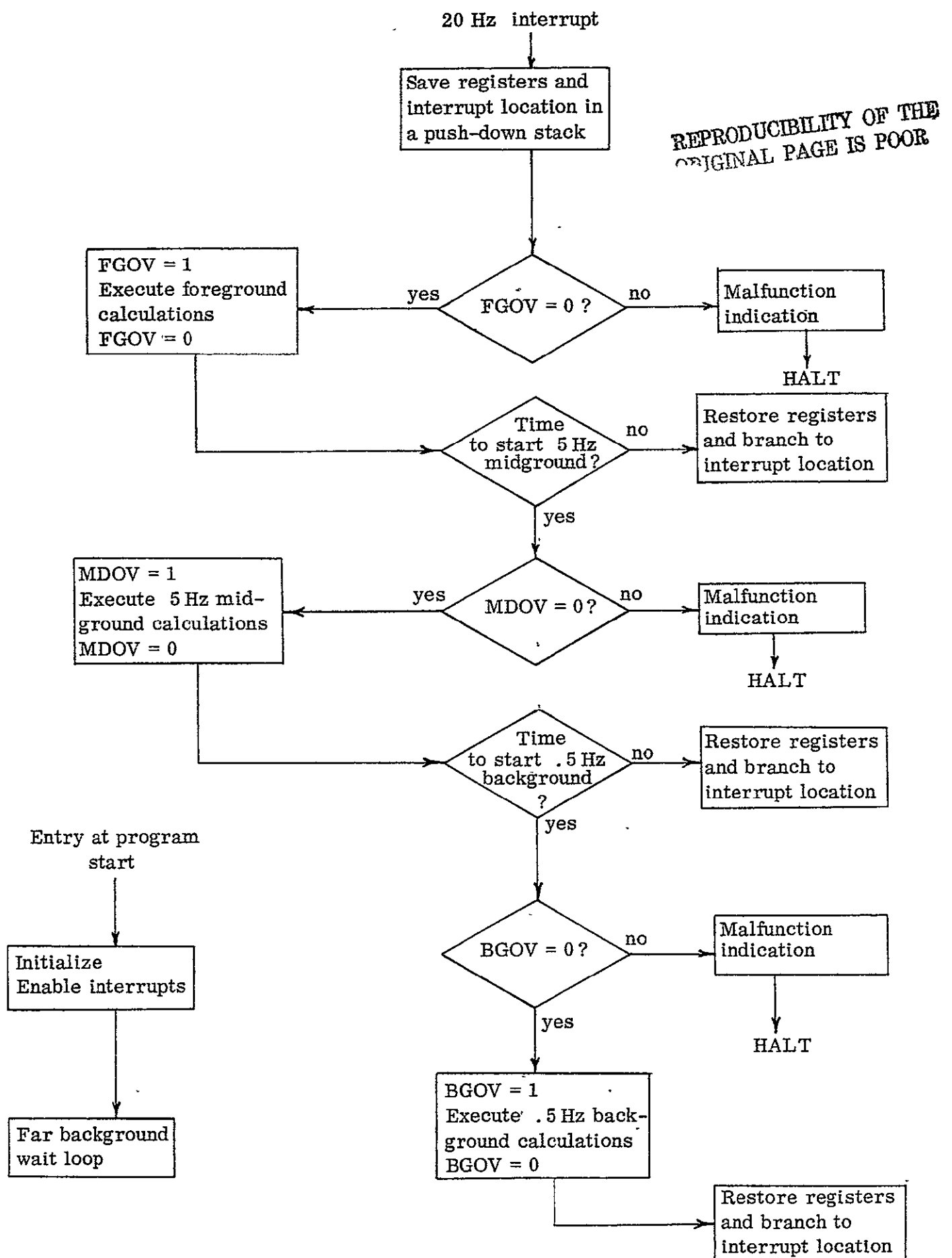


Figure 1. Macro Flow Chart of Executive for Kalman Filter

If it is time to start the .5 Hz calculations, the .5 Hz background overflow flag, BGOV, is tested to determine if a malfunction has occurred. If not, BGOV is set unity and the .5 Hz background calculations initiated. Following completion, BGOV is set zero, the registers are restored, and a branch to the interrupt location made. In this instance, the branch is to a location in the far background wait loop. The executive driver shown in Figure I only executes load, store, increment, and branch-type instructions.

Figure II is a macro flow chart of the foreground calculations executed at a frequency of 20 Hz. The first operation is to bring the body-mounted accelerometer outputs (in the form of delta velocity increments) into the MCP-701. Since these must be transformed to the runway inertial coordinate system, the Euler angles are brought in and corrected by the filter estimates of the gyro drift bias. The Euler transformation is computed and the body accelerations are transformed into the runway system. The acceleration of gravity is added as well as the filter's best estimate of the vertical acceleration bias. The runway accelerations are integrated, using an "average g" concept, to obtain the navigation estimates of position and velocity. The Kalman filter correction of the position and velocity is added to the integrated values using an exponential smoothing technique to avoid discontinuous jumps in the guidance-control system equations.

Following the navigation equation logic, a test is made to determine if a change is ready for input to the navigation equations. This state change will be ready every 2 seconds (after completion of the measurement update logic in the .5 Hz background calculations). If ready, the state change is added to the state estimate carried by the navigation equations, the state change flag set false, and the 20 Hz guidance, control, and display calculations initiated. If not ready, the branch to initiate the 20 Hz guidance, control, and display calculations is made.

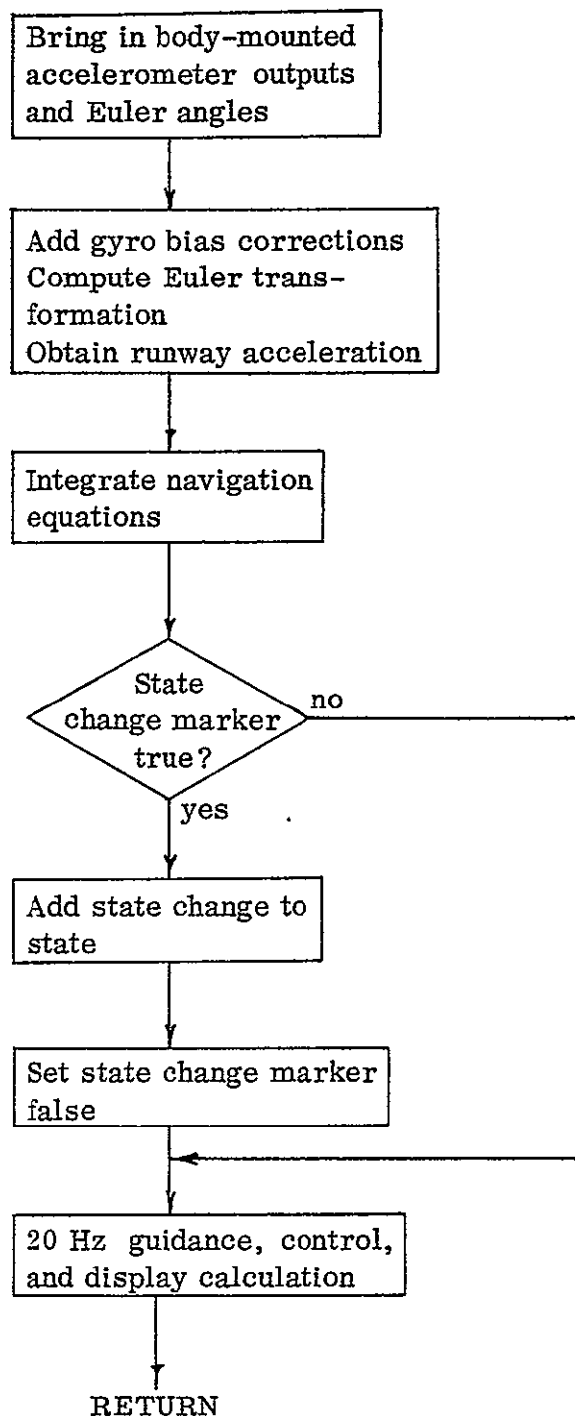


Figure II. Macro Flow Chart of the 20 Hz Calculations
for MCP-701 Computer

Figure III is a macro flow chart of the calculations executed at 5 Hz. The first operation is to update the transition matrix to the current entry time. The residual sums and the partials required for processing these residual sums in the 5. Hz midground logic are then calculated. Validity markers which are set by the receivers are tested to determine if the data is valid. If it is not valid, it is skipped for the particular entry. Following the residual sums, the measurement processing complete marker is tested. This marker is set true in the .5 Hz logic when the measurements (residual sums) have all been processed for the previous .5 second summation interval. If true, the current residual sums are corrected for the calculated incremental state and the state change is set up for input to the system during the subsequent foreground cycle.

After these calculations, the test is made to determine whether the 2-second summation cycle is complete. If true, the residual sums and partials are loaded into arrays for processing in the .5 Hz background logic. The locations for accumulation of residual sums and partial sums are then cleared and the measurement update required marker set true. After completing the above calculations (as required), the program executes the 5 Hz guidance, control, and display logic.

Figure IV is a macro flow chart of the navigation calculations to be executed at a frequency of .5 Hz. The first operation is set to determine if any measurements (residual sums) are ready for processing. The test will generally be passed unless all the measurements were invalid in the 5 Hz midground for the full two-second summation cycle. If residual sums are ready, then the incremental state is calculated for all the available residual sums. Following completion, the measurement processing complete marker is set true. The logic then updates the square root covariance matrix and reduces the square root covariance to upper triangular form with Householder's algorithm. Following this logic, any guidance, control, or display calculations required at the .5 Hz frequency are executed.

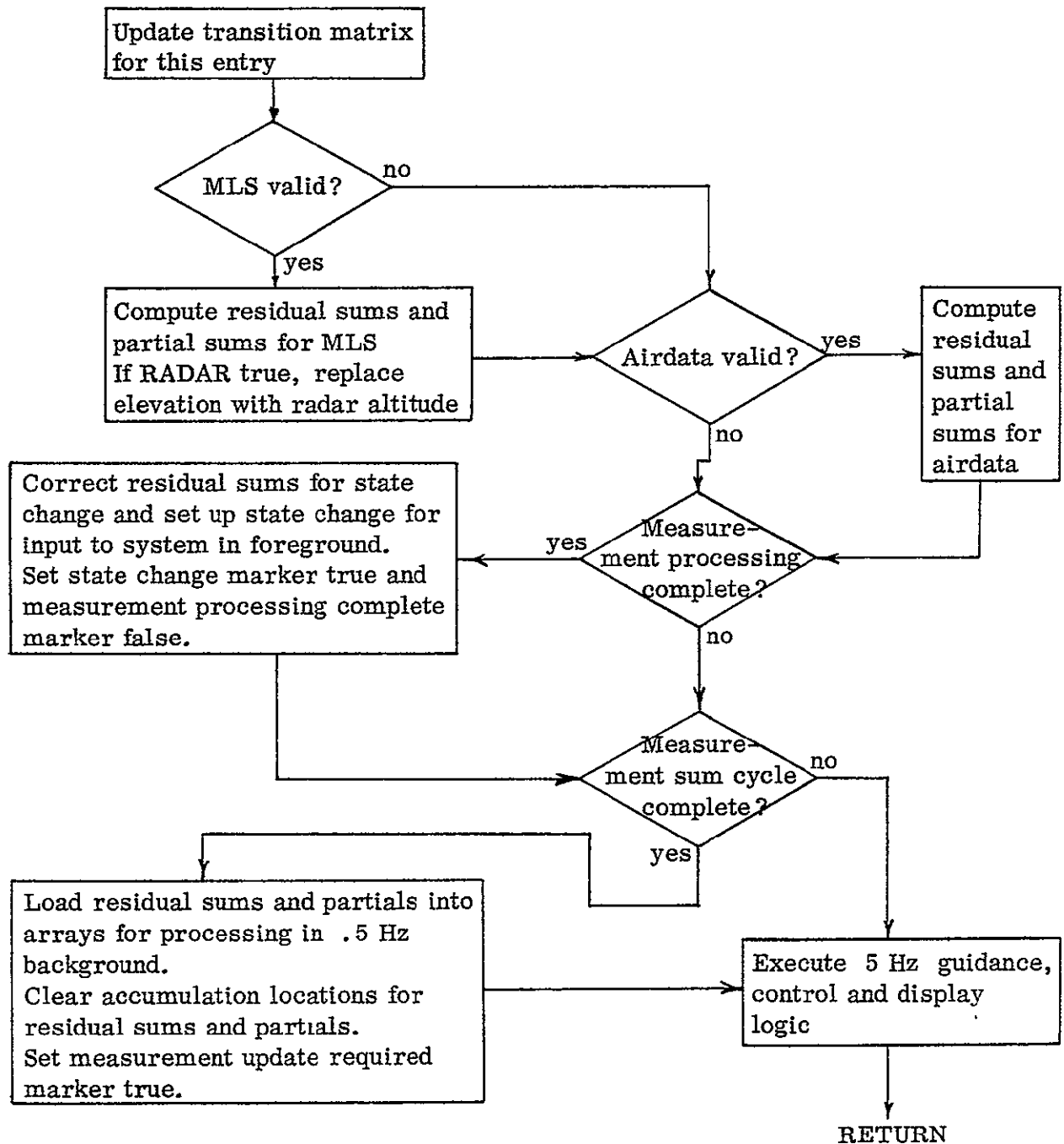


Figure III. Macro Flow Chart of 5 Hz Midground Calculations for MCP-701 Computer

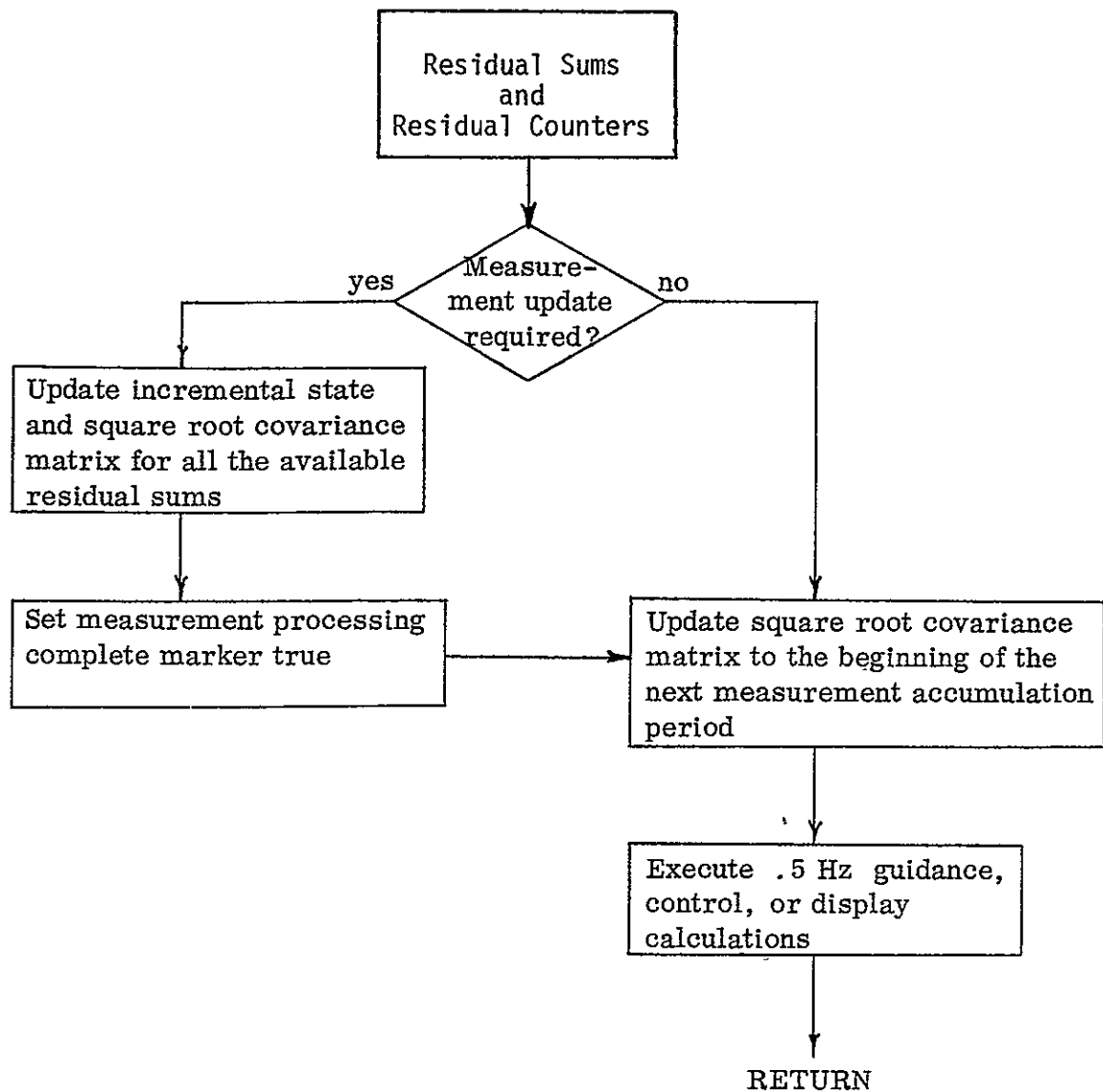


Figure IV. Macro Flow Chart of the .5 Hz Calculations
for the MCP-701 Computer

III. EQUATIONS FOR FOREGROUND COMPUTATIONS

This section contains the equations for the operations outlined in Figure II. Euler angles and body-mounted acceleration outputs are brought in. In order to integrate the accelerations in the runway system and provide the guidance and display information, the following are carried out:

- 1) Get the Euler angles from the INS unit and add the corrections for the gyro drift biases and the runway yaw.

$$\begin{aligned}\text{GYRACC}(1) &= \text{PHI} + \text{XHAT}(5) \\ \text{GYRACC}(2) &= \text{THETA} + \text{XHAT}(6) \\ \text{GYRACC}(3) &= \text{PSI} + \text{XHAT}(7) - \text{PSIR}\end{aligned}\tag{1}$$

- 2) Compute the transformation matrix from body axes to runway axes.

$$\begin{aligned}\text{COST} &= \cos(\text{GYRACC}(2)) \\ \text{SINT} &= \sin(\text{GYRACC}(2)) \\ \text{COSP} &= \cos(\text{GYRACC}(1)) \\ \text{SINP} &= \sin(\text{GYRACC}(1)) \\ \text{COSI} &= \cos(\text{GYRACC}(3)) \\ \text{SINI} &= \sin(\text{GYRACC}(3)) \\ \text{AMXX} &= \text{COST} * \text{COSI} \\ \text{AMXY} &= \text{COST} * \text{SINI} \\ \text{AMXZ} &= -\text{SINT} \\ \text{AMYX} &= -\text{SINI} * \text{COSP} + \text{COSI} * \text{SINP} * \text{SINT} \\ \text{AMYY} &= \text{COSI} * \text{COSP} + \text{SINI} * \text{SINP} * \text{SINT} \\ \text{AMYZ} &= \text{COST} * \text{SINP} \\ \text{AMZX} &= \text{SINI} * \text{SINP} + \text{COSI} * \text{COSP} * \text{SINT} \\ \text{AMZY} &= -\text{COSI} * \text{SINP} + \text{SINI} * \text{COSP} * \text{SINT} \\ \text{AMZZ} &= \text{COST} * \text{COSP}\end{aligned}\tag{2}$$

- 3) Get the body accelerations, $ACC(I)$, and rotate them into the runway axes to obtain the runway specific force components.

$$\begin{aligned} FXYFZ(1) &= AMXX * ACC(1) + AMYX * ACC(2) + AMZX * ACC(3) \\ FXYFZ(2) &= AMXY * ACC(1) + AMYY * ACC(2) + AMZY * ACC(3) \\ FXYFZ(3) &= AMXZ * ACC(1) + AMYZ * ACC(2) + AMZZ * ACC(3) \end{aligned} \quad (3)$$

The specific force vector is stored for use in the state transition matrix in the midground computation loop. To obtain the runway accelerations, we must add the gravity force and the accelerometer bias.

$$\begin{aligned} CACN(1) &= FXYFZ(1) \\ CACN(2) &= FXYFZ(2) \\ CACN(3) &= FXYFZ(3) + G + XHAT(14) \end{aligned} \quad (4)$$

- 4) Integrate the runway acceleration to obtain position and velocity in the runway system.

Interchange the old and new velocity:

$$\begin{aligned} VELO(1) &= VELN(1) \\ VELO(2) &= VELN(2) \\ VELO(3) &= VELN(3) \end{aligned} \quad (5a)$$

Compute the new velocity:

$$\begin{aligned} VELN(1) &= VELO(1) + (CACO(1) + CACN(1)) \Delta T/2 \\ VELN(2) &= VELO(2) + (CACO(2) + CACN(2)) \Delta T/2 \\ VELN(3) &= VELO(3) + (CACO(3) + CACN(3)) \Delta T/2 \end{aligned} \quad (5b)$$

Compute the new position:

$$\begin{aligned} POS(1) &= POS(1) + (VELO(1) + VELN(1)) \Delta T/2 \\ POS(2) &= POS(2) + (VELO(2) + VELN(2)) \Delta T/2 \\ POS(3) &= POS(3) + (VELO(3) + VELN(3)) \Delta T/2 \end{aligned}$$

Interchange the old and new accelerations:

$$\begin{aligned} \text{CACO}(1) &= \text{CACN}(1) \\ \text{CACO}(2) &= \text{CACN}(2) \\ \text{CACO}(3) &= \text{CACN}(3) \end{aligned} \tag{5c}$$

- 5) Compute the smooth estimate of the position and velocity vectors for guidance, control, and display.

$$\begin{aligned} I &= 1, 6 \\ \text{CS}(I) &= \text{CS}(I) * \text{DECAY} \end{aligned} \tag{6}$$

Compute the smooth position:

$$\begin{aligned} I &= 1, 3 \\ \text{XS}(I) &= \text{POS}(I) + \text{CS}(I) \end{aligned} \tag{6a}$$

Compute the smooth velocity:

$$\begin{aligned} J &= I + 3 \\ \text{XS}(J) &= \text{VELN}(J) + \text{CS}(J) \end{aligned}$$

Update the XHAT's for position and velocity:

$$\begin{aligned} \text{XHAT}(1) &= \text{POS}(1) \\ \text{XHAT}(2) &= \text{POS}(2) \\ \text{XHAT}(12) &= \text{POS}(3) \\ \text{XHAT}(3) &= \text{VELN}(1) \\ \text{XHAT}(4) &= \text{VELN}(2) \\ \text{XHAT}(13) &= \text{VELN}(3) \end{aligned} \tag{6b}$$

- 6) Check the ISTATE trigger to see if new Kalman updates are ready for incorporation.

IF (ISTATE.EQ.0) Go to Step 8

- 7) Update XHAT(I) and zero out DX(I)

$$\text{ISTATE} = 0 \tag{7}$$

$$\begin{aligned}
CS(1) &= CS(1) - DX(1) \\
CS(2) &= CS(2) - DX(2) \\
CS(3) &= CS(3) - DX(12) \\
CS(4) &= CS(4) - DX(4) \\
CS(5) &= CS(5) - DX(5) \\
CS(6) &= CS(6) - DX(13)
\end{aligned} \tag{7a}$$

$$\begin{aligned}
I &= 1, 4 \\
XHAT(I) &= XHAT(I) + DX(I) \\
DX(I) &= 0.
\end{aligned} \tag{7b}$$

$$\begin{aligned}
XHAT(12) &= XHAT(12) + DX(12) \\
XHAT(13) &= XHAT(13) + DX(13) \\
DX(12) &= DX(13) = 0.
\end{aligned} \tag{7c}$$

$$\begin{aligned}
CSI &= \cos(GYRACC(3)) \\
SSI &= \sin(GYRACC(3)) \\
CTH &= \cos(GYRACC(2)) \\
STH &= \sin(GYRACC(2))
\end{aligned} \tag{7d}$$

$$X = (CSI * DX(5) + SSI * DX(6)) / CTH \tag{7e}$$

$$\begin{aligned}
XHAT(5) &= XHAT(5) + X \\
XHAT(6) &= XHAT(6) + CSI * DX(6) - SSI * DX(5) \\
XHAT(7) &= XHAT(7) + DX(7) + X * STH \\
DX(5) &= DX(6) = DX(7) = 0.
\end{aligned} \tag{7f}$$

Update the remaining XHAT's

$$\begin{aligned}
I &= 8, 11 \\
XHAT(I) &= XHAT(I) + DX(I) \\
DX(I) &= 0.
\end{aligned} \tag{7g}$$

$$\begin{aligned}
XHAT(14) &= XHAT(14) + DX(14) \\
XHAT(15) &= XHAT(15) + DX(15) \\
DX(14) &= DX(15) = 0.
\end{aligned} \tag{7h}$$

8) Increase the foreground counter by unity.

IFIL = IFIL + 1

9) Check if the midground logic is to be initiated.

IF (MOD(IFIL,MIDGR).EQ.0) Go to Step 1 of midground
computations and set IFIL = 0

This completes the foreground logic. Restore registers and branch to
interrupt locations.

IV. EQUATIONS FOR MIDGROUND COMPUTATIONS

The midground loop is executed once every MIDGR entry of the foreground loop. Thus, if $MIDGR = 4$ and the foreground loop is 20 Hz, the midground loop is 5 Hz. The midground logic is designed to compute the residuals between the navigational aid measurements and the filter estimates of the measurements, and sum the residuals. In addition, this section computes the partial derivatives of the observations with respect to the solved-for variables and sums the partials. It updates the state transition matrix and, when the background trigger is set ready for entry, it loads the sums of the residuals, the number of observations in each sum, and the sum of the partials into an array for use in the background Kalman update loop. It then zeros out the sum arrays and starts over again.

The equations for the specific steps are given as follows:

1) Update state transition matrix

$$\begin{aligned} A(4) &= F_{X F Y F Z}(3) * A(1) \\ A(5) &= - F_{X F Y F Z}(2) * A(1) \\ A(6) &= F_{X F Y F Z}(1) * A(1) \\ A(7) &= A(4) * A(2) \\ A(8) &= A(5) * A(2) \\ A(9) &= A(6) * A(2) \\ AP(1) &= AP(1) + A(1) \\ AP(2) &= AP(2) + A(7) \\ AP(3) &= AP(3) + A(8) \\ AP(4) &= AP(4) + A(1) \\ AP(5) &= AP(5) - A(7) \\ AP(6) &= AP(6) + A(9) \\ AP(7) &= AP(7) + A(4) \end{aligned} \tag{8}$$

$$\begin{aligned}
AP(8) &= AP(8) + A(5) \\
AP(9) &= AP(9) - A(4) \\
AP(10) &= AP(10) + A(6) \\
AP(11) &= AP(11) + A(10) \\
AP(12) &= AP(12) + A(11) \\
AP(13) &= AP(13) + A(12) \\
AP(14) &= AP(14) + A(13) \\
AP(15) &= AP(15) + A(14) \\
AP(16) &= AP(16) + A(15) \\
AP(17) &= AP(17) + A(16) \\
AP(18) &= AP(18) - A(8) \\
AP(19) &= AP(19) - A(9) \\
AP(20) &= AP(20) + A(1) \\
AP(21) &= AP(21) + A(3) \\
AP(22) &= AP(22) - A(5) \\
AP(23) &= AP(23) - A(6) \\
AP(24) &= AP(24) + A(1) \\
AP(25) &= AP(25) + A(17) \\
AP(26) &= AP(26) + A(18)
\end{aligned}
\tag{8} \text{ cont.}$$

- 2) Compute the coordinates of the aircraft relative in the MLS coordinate system.

$$\begin{aligned}
&\text{For azimuth and range:} \\
XXN &= -POS(1) + XAZ(1) \\
YYN &= POS(2) \\
ZZN &= -POS(3) + XAZ(3)
\end{aligned}
\tag{9}$$

$$\begin{aligned}
&\text{For elevation, we have:} \\
XEN &= XXN - XXO \\
YEN &= YYN - YYO \\
ZEN &= ZZN - ZZO
\end{aligned}
\tag{9a}$$

Compute the range:

$$\begin{aligned} \text{SRAN} &= \text{XXN}^2 + \text{ZZN}^2 \\ \text{XRANGE} &= (\text{SRAN} + \text{YYN}^2)^{1/2} \end{aligned} \quad (9b)$$

- 3) Get azimuth measurement and check validity.

$$\text{AZIMUX} = 0.$$

IF (.NOT.VALID(1)) Go to next measurement, Step 9.

- 4) If valid, compute AZIMUX and residual

$$\begin{aligned} \text{AZIMUX} &= \text{Sin}^{-1} (-\text{YYN}/\text{XRANGE}) \\ \text{YRA} &= \text{AZIMUT} - \text{AZIMUX} \end{aligned} \quad (10)$$

- 5) Check residual; if too large, proceed to Step 9.

IF (ABS(YRA).GT.TUMUCH(1)) Go to next measurement,
Step 9.

- 6) Compute sum of residuals and number of measurements in sum:

$$\begin{aligned} \text{SRES}(1) &= \text{SRES}(1) + \text{YRA} \\ \text{NRES}(1) &= \text{NRES}(1) + 1 \end{aligned} \quad (11)$$

- 7) Compute partials of azimuth:

$$\begin{aligned} \text{SQRAN} &= 1./(\text{SRAN})^{1/2} \\ \text{B} &= \text{YYN}/\text{XRANGE} * \text{SQRAN} \\ \text{UA}(1) &= -\text{B} * \text{XXN} \\ \text{UA}(2) &= -\text{SQRAN} + \text{B} * \text{YYN} \\ \text{UA}(3) &= -\text{B} * \text{ZZN} \end{aligned} \quad (12)$$

- 8) Accumulate the sums of products of the state transition matrix into the vector of the partials. The state transition matrix is represented as a vector AP(26).

$$\begin{aligned}
 \text{HRES1}(1) &= \text{HRES1}(1) + \text{UA}(1) \\
 \text{HRES1}(2) &= \text{HRES1}(2) + \text{UA}(2) \\
 \text{HRES1}(3) &= \text{HRES1}(3) + \text{UA}(1) * \text{AP}(1) \\
 \text{HRES1}(4) &= \text{HRES1}(4) + \text{UA}(2) * \text{AP}(4) \\
 \text{HRES1}(5) &= \text{HRES1}(5) + \text{UA}(2) * \text{AP}(5) + \text{UA}(3) * \text{AP}(18) \\
 \text{HRES1}(6) &= \text{HRES1}(6) + \text{UA}(1) * \text{AP}(2) + \text{UA}(3) * \text{AP}(19) \\
 \text{HRES1}(7) &= \text{HRES1}(7) + \text{UA}(1) * \text{AP}(3) + \text{UA}(2) * \text{AP}(6) \\
 \text{HRES1}(12) &= \text{HRES1}(12) + \text{UA}(3) \\
 \text{HRES1}(13) &= \text{HRES1}(13) + \text{UA}(3) * \text{AP}(20) \\
 \text{HRES1}(14) &= \text{HRES1}(14) + \text{UA}(3) * \text{AP}(21)
 \end{aligned} \tag{13}$$

- 9) Get elevation measurement. If RADAR is true, proceed to Step 15.
Check validity of elevation measurement:

ELEVAX = 0.
IF (.NOT.VALID(2)) Go to next measurement type, Step 15.

- 10) If valid, compute ELEVAX and residual:

$$\begin{aligned}
 \text{XYN} &= \text{XEN}^2 + \text{YEN}^2 \\
 \text{REV} &= \text{XYN} + \text{ZEN}^2 \\
 \text{SXYN} &= 1. / (\text{XYN})^{1/2} \\
 \text{ZRET} &= \text{ZEN} * \text{SXYN} / \text{REV} \\
 \text{ELEVAX} &= \text{Sin}^{-1} (\text{ZEN} / (\text{REV})^{1/2}) \\
 \text{YRE} &= \text{ELEVAT} - \text{ELEVAX}
 \end{aligned} \tag{14}$$

- 11) Check residual; if too large, go to Step 15.

IF (ABS(YRE).GT.TUMUCH(2)) Go to Step 15.

- 12) Compute sum of residuals and number of measurements in sum:

$$\begin{aligned} \text{SRES}(2) &= \text{SRES}(2) + \text{YRE} \\ \text{NRES}(2) &= \text{NRES}(2) + 1 \end{aligned} \tag{15}$$

- 13) Compute partials of elevation:

$$\begin{aligned} \text{UE}(1) &= \text{XEN} * \text{ZRET} \\ \text{UE}(2) &= - \text{YEN} * \text{ZRET} \\ \text{UE}(3) &= \text{ZEN} * \text{ZRET} - \text{SXYN} \end{aligned} \tag{16}$$

- 14) Accumulate the sums of the partials, as in Step 8:

$$\begin{aligned} \text{HRES2}(1) &= \text{HRES2}(1) + \text{UE}(1) \\ \text{HRES2}(2) &= \text{HRES2}(2) + \text{UE}(2) \\ \text{HRES2}(3) &= \text{HRES2}(3) + \text{UE}(1) * \text{AP}(1) \\ \text{HRES2}(4) &= \text{HRES2}(4) + \text{UE}(2) * \text{AP}(4) \\ \text{HRES2}(5) &= \text{HRES2}(5) + \text{UE}(2) * \text{AP}(5) + \text{UE}(3) * \text{AP}(18) \\ \text{HRES2}(6) &= \text{HRES2}(6) + \text{UE}(1) * \text{AP}(2) + \text{UE}(3) * \text{AP}(19) \\ \text{HRES2}(7) &= \text{HRES2}(7) + \text{UE}(1) * \text{AP}(3) + \text{UE}(2) * \text{AP}(6) \\ \text{HRES2}(12) &= \text{HRES2}(12) + \text{UE}(3) \\ \text{HRES2}(13) &= \text{HRES2}(13) + \text{UE}(3) * \text{AP}(20) \\ \text{HRES2}(14) &= \text{HRES2}(14) + \text{UE}(3) * \text{AP}(21) \end{aligned} \tag{17}$$

- 15) Get range measurement and check validity:

$\text{ERANGE} = 0.$
IF (.NOT.VALID(3)) Go to next measurement, Step 21.

- 16) If valid, compute ERANGE and residual:

$$\begin{aligned} \text{ERANGE} &= \text{XRANGE} \\ \text{YRR} &= \text{RANGE} - \text{ERANGE} \end{aligned} \tag{18}$$

17) Check residual; if too large, go to Step 21.

IF (ABS(YRR).GT.TUMUCH(3)) Go to Step 21.

18) Compute the sum of residuals and the number of measurements in the sum:

$$\begin{aligned} \text{SRES}(3) &= \text{SRES}(3) + \text{YRR} \\ \text{NRES}(3) &= \text{NRES}(3) + 1 \end{aligned} \tag{19}$$

19) Compute partials of range:

$$\begin{aligned} \text{UR}(1) &= -\text{XXN}/\text{ERANGE} \\ \text{UR}(2) &= \text{YYN}/\text{ERANGE} \\ \text{UR}(3) &= -\text{ZZN}/\text{ERANGE} \end{aligned} \tag{20}$$

20) Accumulate sum of range partials, as in Step 8:

$$\begin{aligned} \text{HRES3}(1) &= \text{HRES3}(1) + \text{UR}(1) \\ \text{HRES3}(2) &= \text{HRES3}(2) + \text{UR}(2) \\ \text{HRES3}(3) &= \text{HRES3}(3) + \text{UR}(1) * \text{AP}(1) \\ \text{HRES3}(4) &= \text{HRES3}(4) + \text{UR}(2) * \text{AP}(4) \\ \text{HRES3}(5) &= \text{HRES3}(5) + \text{UR}(2) * \text{AP}(5) + \text{UR}(3) * \text{AP}(18) \\ \text{HRES3}(6) &= \text{HRES3}(6) + \text{UR}(1) * \text{AP}(2) + \text{UR}(3) * \text{AP}(19) \\ \text{HRES3}(7) &= \text{HRES3}(7) + \text{UR}(1) * \text{AP}(3) + \text{UR}(2) * \text{AP}(6) \\ \text{HRES3}(12) &= \text{HRES3}(12) + \text{UR}(3) \\ \text{HRES3}(13) &= \text{HRES3}(13) + \text{UR}(3) * \text{AP}(20) \\ \text{HRES3}(14) &= \text{HRES3}(14) + \text{UR}(3) * \text{AP}(21) \end{aligned} \tag{21}$$

21) If radar trigger is false, go to Step 27. If radar is true, get radar observation and check validity:

ALT = 0.

IF (.NOT.RADAR) Go to Step 27.

IF (.NOT.VALID(4)) Go to Step 27.

22) Compute altitude and residual:

$$\begin{aligned} \text{ALT} &= -\text{POS}(3) \\ \text{YRAL} &= \text{ALTU} - \text{ALT} \end{aligned} \tag{22}$$

23) Check residual; if too large, go to Step 27:

IF (ABS(YRAL).GT.TUMUCH(4)) Go to Step 27.

24) Compute sum of residuals and number of measurements in sum:

$$\begin{aligned} \text{SRES}(2) &= \text{SRES}(2) + \text{YRAL} \\ \text{NRES}(2) &= \text{NRES}(2) + 1 \end{aligned} \tag{23}$$

25) Compute partials of altitude:

$$\text{UAR} = -1. \tag{24}$$

26) Accumulate sum of partials:

$$\begin{aligned} \text{HRES2}(5) &= \text{HRES2}(5) + \text{UAR} * \text{AP}(18) \\ \text{HRES2}(6) &= \text{HRES2}(6) + \text{UAR} * \text{AP}(19) \\ \text{HRES2}(12) &= \text{HRES2}(12) + \text{UAR} \\ \text{HRES2}(13) &= \text{HRES2}(13) + \text{UAR} * \text{AP}(20) \\ \text{HRES2}(14) &= \text{HRES2}(14) + \text{UAR} * \text{AP}(21) \end{aligned} \tag{25}$$

27) Get airspeed measurement and compute airspeed estimate and residual:

$$\begin{aligned} \text{AIR} &= 0. \\ \text{VX} &= \text{VELN}(1) - \text{XHAT}(11) \\ \text{VY} &= \text{VELN}(2) - \text{XHAT}(12) \\ \text{VZ} &= \text{VELN}(3) \\ \text{AIR} &= (\text{VX}^2 + \text{VY}^2 + \text{VZ}^2)^{1/2} \\ \text{YRAR} &= \text{AIRSP} - \text{AIR} \end{aligned} \tag{26}$$

28) Check residual; if too large, go to Step 32:

IF (ABS(YRAR).GT. TUMUCH(5)) Go to Step 32.

29) Compute sum of residuals and number of measurements in sum:

$$\begin{aligned} \text{SRES}(4) &= \text{SRES}(4) + \text{YRAR} \\ \text{NRES}(4) &= \text{NRES}(4) + 1 \end{aligned} \quad (27)$$

30) Compute partials of airspeed:

$$\begin{aligned} \text{UAI}(1) &= -\text{VX}/\text{AIR} \\ \text{UAI}(2) &= -\text{VY}/\text{AIR} \end{aligned} \quad (28)$$

31) Accumulate sum of partials:

$$\begin{aligned} \text{HRES4}(10) &= \text{HRES4}(10) + \text{UAI}(1) * \text{AP}(16) \\ \text{HRES4}(11) &= \text{HRES4}(11) + \text{UAI}(2) * \text{AP}(17) \end{aligned} \quad (29)$$

32) Check to see if background has corrected state (is MPROC = 1 ?).
If not, proceed to Step 36.

IF (MPROC.EQ.0) Go to Step 36.

33) Correct newly started residual sums for Kalman filter state corrections that have not yet been incorporated into estimated observations:

$$\begin{aligned} I &= 1, 4 \\ \text{SRES}(I) &= \text{SRES}(I) - \sum_{j=1}^{15} \text{HRESI}(J) * \text{DX}(J) \end{aligned} \quad (30)$$

NOTE: HRESI(J) is HRES1(J) or HRES2(J) or etc.

34) Propagate state corrections, DX(I), to next filter time step:

DATA NR = 3, 6, 7, 4, 5, 7, 6, 7, 5, 7, 5, 6, 7, 8, 9, 10, 11, 5,
6, 13, 14, 5, 6, 14, 14, 15

DATA NE = 1, 4, 7, 9, 11, 12, 13, 14, 15, 16, 17, 18, 22, 25,
26, 27

I = 1, 15

M = NE(I)

N = NE(I+1)-1

$$S(I) = \sum_{L=M}^N AP(L) * DX(NR(L)) \quad (31)$$

After all S(I) are computed, set DX(I) equal to S(I):

I = 1, 15

$$DX(I) = S(I) \quad (32)$$

35) Set trigger to incorporate DX(I) into XHAT in foreground computations:

$$ISTATE = 1 \quad (33)$$

36) Check measurement count trigger for background computation cycle:

IF (MESCNT.GT.0) Go to Step 40.

37) If the proper number of measurement step intervals have transpired, load residual sums and partial sums into registers and zero out the old registers:

I = 1, 4

$$IND(I) = NRES(I) \quad (34)$$

$$NRES(I) = 0$$

YR(I) = SRES(I)

SRES(I) = 0.

I = 1,15

H1(I) = HRES1(I)

HRES1(I) = 0.

H2(I) = HRES2(I)

HRES2(I) = 0.

H3(I) = HRES3(I)

HRES3(I) = 0.

H4(I) = HRES4(I)

HRES4(I) = 0.

I = 1,26

APS(I) = AP(I)

AP(I) = 0.

(34) cont.

38) Check for switch over to RADAR altimeter:

IF (RADAR.OR.-POS(3).GT.HRADAR) Go to Step 40.

39) Set constants for RADAR processing:

TUMUCH(2) = RDMUCH

STDI(2) = STRAD

RADAR = TRUE

(35)

40) Increase measurement interval counter by unity:

MESCNT = MESCNT + 1

41) Check to see if background computations are to be initiated:

IF (MESCNT.EQ.INTIME-1) Go to Background Step 1.

.42) This completes the midground computations loop. Restore registers and proceed to the interrupt logic.

V. EQUATIONS FOR BACKGROUND COMPUTATIONS

The background computations are executed every time that MESCNT equals zero. In this cycle, the changes in the state, DX(I), are updated, the square root covariance matrix, WT(15,15), is updated, the diagonal process noise matrix is added to WT(15,15) to produce a 15 x 27 matrix, the Householder routine is employed to transform the 15 x 27 matrix into a triangular matrix, WT(15,15), and finally the state correction, DX(I), and WT(15,15) are propagated to proper time by the state transition matrix, APS(I).

The equations for the background logic are as follows:

- 1) Check if number of azimuth observations is greater than zero:

IF (IND(1).LE.0) Go to Step 3.

- 2) Compute the Potter solution for the Kalman filter using the square root formulation:

$$I = 1, 15$$

$$D(I) = \sum_{J=1}^{15} H1(J) * WT(I, J) \quad (36)$$

$$Q(1) = STDI(1)^2 * FLOAT(IND(1)) \quad (36a)$$

$$VAR(1) = Q(1) + \sum_{I=1}^{15} D(I)^2 \quad (36b)$$

$$E(1) = (YR(1) - \sum_{I=1}^{15} H1(I) * DX(I)) / VAR(1) \quad (36c)$$

$$AD(1) = VAR(1) + VAR(1) * (Q(1)/VAR(1))^{1/2} \quad (36d)$$

$$I = 1, 15$$

$$BD(I) = \sum_{J=1}^{15} WT(J, I) * D(J) \quad (36e)$$

$$\begin{aligned}
I &= 1,15 \\
DX(I) &= DX(I) + BD(I) * E(1)
\end{aligned} \tag{36f}$$

$$\begin{aligned}
I &= 1,15 \\
J &= 1,15 \\
WT(J,I) &= WT(J,I) - D(J) * BD(I)/AD(1)
\end{aligned} \tag{36g}$$

- 3) Check if number of elevation (or radar altimeter, if RADAR is true) measurements is greater than zero;

IF (IND(2).LE.0) Go to Step 5.

- 4) Compute Potter solution for the Kalman filter, as in Step 2:

$$\begin{aligned}
I &= 1,15 \\
D(I) &= \sum_{J=1}^{15} H2(J) * WT(I,J)
\end{aligned} \tag{37}$$

$$Q(2) = STDI(2)^2 * FLOAT(IND(2)) \tag{37a}$$

$$VAR(2) = Q(2) + \sum_{I=1}^{15} D(I)^2 \tag{37b}$$

$$E(2) = (YR(2) - \sum_{I=1}^{15} H2(I) * DX(I))/VAR(2) \tag{37c}$$

$$AD(2) = VAR(2) + VAR(2) * (Q(2)/VAR(2))^{1/2} \tag{37d}$$

$$\begin{aligned}
I &= 1,15 \\
BD(I) &= \sum_{J=1}^{15} WT(J,I) * D(J).
\end{aligned} \tag{37e}$$

$$\begin{aligned}
I &= 1,15 \\
DX(I) &= DX(I) + BD(I) * E(2)
\end{aligned} \tag{37f}$$

$$\begin{aligned}
I &= 1, 15 \\
J &= 1, 15 \\
WT(J, I) &= WT(J, I) - D(J) * BD(I) / AD(2)
\end{aligned} \tag{37g}$$

5) Check if number of range measurements is greater than zero:

IF (IND(3).LE.0) Go to Step 7.

6) Compute Potter solution for the Kalman filter, as in Step 2:

$$\begin{aligned}
I &= 1, 15 \\
D(I) &= \sum_{J=1}^{15} H3(J) * WT(I, J)
\end{aligned} \tag{38}$$

$$Q(3) = STDI(3)^2 + FLOAT(IND(3)) \tag{38a}$$

$$VAR(3) = Q(3) + \sum_{I=1}^{15} D(I)^2 \tag{38b}$$

$$E(3) = (YR(3) - \sum_{I=1}^{15} H3(I) * DX(I)) / VAR(3) \tag{38c}$$

$$AD(3) = VAR(3) + VAR(3) * (Q(3) / VAR(3))^{1/2} \tag{38d}$$

$$\begin{aligned}
I &= 1, 15 \\
BD(I) &= \sum_{J=1}^{15} WT(J, I) * D(J)
\end{aligned} \tag{38e}$$

$$\begin{aligned}
I &= 1, 15 \\
DX(I) &= DX(I) + BD(I) * E(3)
\end{aligned} \tag{38f}$$

$$\begin{aligned}
I &= 1, 15 \\
J &= 1, 15 \\
WT(J, I) &= WT(J, I) - D(J) * BD(I) / AD(3)
\end{aligned} \tag{38g}$$

7) Check if number of airspeed measurements is greater than zero:

IF (IND(4).LE.0) Go to Step 9

8) Compute Potter solution for Kalman filter, as in Step 2:

$$I = 1, 15$$

$$D(I) = \sum_{J=1}^{15} H4(J) * WT(I, J) \quad (39)$$

$$Q(4) = STDI(4)^2 * FLOAT(IND(4)) \quad (39a)$$

$$VAR(4) = Q(4) + \sum_{I=1}^{15} D(I)^2 \quad (39b)$$

$$E(4) = (YR(4) - \sum_{I=1}^{15} H4(I) * DX(I)) / VAR(4) \quad (39c)$$

$$AD(4) = VAR(4) + VAR(4) * (Q(4)/VAR(4))^{1/2} \quad (39d)$$

$$I = 1, 15$$

$$BD(I) = \sum_{J=1}^{15} WT(J, I) * D(J) \quad (39e)$$

$$I = 1, 15$$

$$DX(I) = DX(I) + BD(I) * E(4) \quad (39f)$$

$$I = 1, 15$$

$$J = 1, 15$$

$$WT(J, I) = WT(J, I) - D(J) * BD(I) / AD(4) \quad (39g)$$

9) Propagate the new DX(I) solution to the proper time for incorporation into the guidance and display equations:

DATA NR = 3, 6, 7, 4, 5, 7, 6, 7, 5, 7, 5, 6, 7, 8, 9, 10, 11, 5,
6, 13, 14, 5, 6, 14, 14, 15

DATA NE = 1, 4, 7, 9, 11, 12, 13, 14, 15, 16, 17, 18, 22, 25,
26, 27

I = 1,15

M = NE(I)

N = NE(I+1)-1

$$SS(I) = \sum_{L=M}^N APS(L) * DX(NR(L)) \quad (40)$$

I = 1,15

$$DX(I) = SS(I) \quad (40a)$$

10) Propagate the covariance matrix, WT(J,I) to the proper time:

DATA NR = 3, 6, 7, 4, 5, 7, 6, 7, 5, 7, 5, 6, 7, 8, 9, 10, 11, 5,
6, 13, 14, 5, 6, 14, 14, 15

DATA NE = 1, 4, 7, 9, 11, 12, 13, 14, 15, 16, 17, 18, 22, 25,
26, 27

Do through (41a)

J = 1,15

I = 1,15

M = NE(I)

N = NE(I+1)-1

$$WW(I) = \sum_{L=M}^N APS(L) * WT(J, NR(L)) \quad (41)$$

I = 1,15

$$WT(J,I) = WW(I) \quad (41a)$$

11) Add process noise matrix, B(I), to WT(15,15) to produce WT(27,15):

Do through (42)

$$I = 1, 9$$

$$J = I+2$$

$$K = I+15$$

$$WT(K, J) = B(I) \quad (42)$$

Add the process noise for the last three rows:

$$WT(25, 13) = B(10)$$

$$WT(26, 14) = B(11) \quad (43)$$

$$WT(27, 15) = B(12)$$

- 12) Apply the Householder method to WT(27, 15) to produce triangular matrix WT(15, 15):

Do through (44j)

$$I = 1, 15$$

$$K = I, 27$$

$$DD(K) = WT(K, I) \quad (44)$$

Compute the sum of squares:

$$SM(I) = \sum_{L=I}^{27} DD(L)^2 \quad (44a)$$

Compute the variable DSIG(I):

$$DSIG(I) = \text{SIGN}(DD(I))(SM(I))^{1/2} \quad (44b)$$

Compute the variable BETU(I):

$$BETU(I) = 1.0 / (SM(I) + DSIG(I) * DD(I)) \quad (44c)$$

Redefine DD(I) to be:

$$DD(I) = DD(I) + DSIG(I) \quad (44d)$$

Compute the vector WS(M) as a matrix vector product:

$$\begin{aligned} M &= I+1, 15 \\ WS(M) &= \left(\sum_{N=I+1}^{27} WT(M, N) * DD(N) \right) * BETU(I) \end{aligned} \quad (44e)$$

For each I, form the new WT(J, L) matrix:

$$J = 1, 27$$

$$L = 1, 15$$

For J < I:

$$WT(J, L) = WT(J, L) \quad (44f)$$

For J=I and L < I:

$$WT(I, L) = WT(I, L) \quad (44g)$$

For J=I and L=I:

$$WT(I, I) = -DSIG(I) \quad (44h)$$

For J=I and L > I:

$$WT(I, L) = 0. \quad (44i)$$

For J > I:

$$WT(J, L) = WT(J, L) - WS(J) * DD(L) \quad (44j)$$

When the process is complete, WT(J, I) will be a lower triangular square matrix.

13) Set MPROC = 1 to carry out Step 33 in midground loop.

This completes the background computations. Reset the registers and return to the interrupt logic.

VI. INITIALIZATION

This section contains the initial values of the variables, program constants, and set zero that must be specified prior to carrying out the normal program outlined in the foreground, midground, and background loops.

1) Program Integers and Logicals:

```
MPROC   =  0
ISTATE  =  0
RADAR   =  .FALSE.
IFIL    =  0
MIDGR   =  4
INTIME  = 10
MESCNT  =  0
VALID(I) = .TRUE.           I = 1, 4
IND(I)   =  0               I = 1, 4
NRES(I)  =  0               I = 1, 4
```

2) Variables to be set to zero:

```
XHAT(I) = 0.               I = 1, 15
XS(I)    = 0.               I = 1, 6
CS(I)    = 0.               I = 1, 6
DX(I)    = 0.               I = 1, 15
AP(I)    = 0.               I = 1, 26
APS(I)   = 0.               I = 1, 26
HRES1(I) = 0.               I = 1, 15
HRES2(I) = 0.               I = 1, 15
HRES3(I) = 0.               I = 1, 15
HRES4(I) = 0.               I = 1, 15
YR(I)    = 0.               I = 1, 4
```

SRES(I)	=	0.	I = 1, 4
WT(I, J)	=	0.	I = 1, 27; J = 1, 15
H1(I)	=	0.	I = 1, 15
H2(I)	=	0.	I = 1, 15
H3(I)	=	0.	I = 1, 15
H4(I)	=	0.	I = 1, 15
YRA	=	0.	
YRE	=	0.	
YRR	=	0.	
YRAL	=	0.	

3) Nonvariable Program Parameters to be Stored:

DTIME	=	.05	
DECAY	=	.9753	
PI	=	3.1416	
TPI	=	2. * PI	
KNOT	=	1.687	
NMILE	=	6076.	
RADIAN	=	180./PI	
DTFILT	=	FLOAT(MIDGR) * DTIME	
DLTT	=	DTFILT * FLOAT(INTIME)	
DT2FIL	=	DTFILT/2.	
HRADAR	=	100.	Maximum radar altitude
PSIR	=	(Runway yaw angle in degrees) * RADIAN	
XAZ(1)	=	X location of azimuth antenna	
XAZ(2)	=	Y location of azimuth antenna	
XAZ(3)	=	Z location of azimuth antenna	
XEL(1)	=	X location of elevation antenna	
XEL(2)	=	Y location of elevation antenna	
XEL(3)	=	Z location of elevation antenna	

XXO	=	XAZ(1) - XEL(1)
YYO	=	XAZ(2) - XEL(2)
ZZO	=	XAZ(3) - XEL(3)
SIGNO(1)	=	25. $\dot{x}$ process noise parameter
SIGNO(2)	=	25. $\dot{y}$ process noise parameter
SIGNO(10)	=	25. $\dot{z}$ process noise parameter
SIGNO(3)	=	.01 p gyro process noise parameter
SIGNO(4)	=	.01 q gyro process noise parameter
SIGNO(5)	=	.01 r gyro process noise parameter
SIGNO(11)	=	.01 $\ddot{z}$ process noise parameter
SIGNO(8)	=	10. x wind process noise parameter
SIGNO(9)	=	10. y wind process noise parameter
SIGNO(6)	=	500. VORTAC range process noise parameter
SIGNO(7)	=	.03 VORTAC bearing process noise parameter
SIGNO(12)	=	200. Baro-altimeter process noise parameter
TAU(I)	=	100. I = 1, 9
UNID	=	2. * DLTT
RDMUCH	=	100,000. Radar altitude edit
TUMUCH(1)	=	.003 Azimuth edit
TUMUCH(2)	=	.003 Elevation edit
TUMUCH(3)	=	150. Range edit
TUMUCH(4)	=	RDMUCH
TUMUCH(5)	=	20. Airspeed edit
STDI(1)	=	.02 Azimuth sigma, radians
STDI(2)	=	.02 Elevation sigma, radians
STDI(3)	=	15. Range sigma, ft.
STDI(4)	=	20. Airspeed sigma, ft./sec.
STDRAD	=	3. Radar altimeter sigma, ft.

WT(8, 8)	=	SIG(6)
WT(9, 9)	=	SIG(7)
WT(10, 10)	=	SIG(5)
WT(11, 11)	=	SIG(5)
WT(14, 14)	=	SIG(4)
WT(15, 15)	=	SIG(8)
POS(1)	=	Best X position in runway coordinates
POS(2)	=	Best Y position in runway coordinates
POS(3)	=	Best Z position in runway coordinates
VELN(1)	=	Best $\dot{X}$ velocity in runway coordinates
VELN(2)	=	Best $\dot{Y}$ velocity in runway coordinates
VELN(3)	=	Best $\dot{Z}$ velocity in runway coordinates
XHAT(10)	=	Best wind X estimate
XHAT(11)	=	Best wind Y estimate
XHAT(1)	=	POS(1)
XHAT(2)	=	POS(2)
XHAT(12)	=	POS(3)
XHAT(3)	=	VELN(1)
XHAT(4)	=	VELN(2)
XHAT(13)	=	VELN(3)

VII. STORAGE AND SCALING

This section contains the number of storage cells required for the various phases of the Kalman filter. The details are listed in tabular form. Each table contains the type of variable, name of the variable, origin (where it is computed), and the minimum and maximum values of each variable. Finally, each table contains a count of the number of items, by type, required for the computation of that phase.

The variable types are:

V	=	variable	(XHAT(1), etc.)
P	=	fixed parameter	(G, etc.)
S	=	scratch variable for intermediate computation	(ZRET, etc.)
I	=	integer	(IFIL, etc.)
L	=	logical	(RADAR, etc.)

The variable names are self-explicit. The origin column describes where the variable is computed, and defines the interface areas. The various origins are:

FG	=	foreground loop
MG	=	midground loop
BG	=	background loop
INITIAL	=	initialization
INS	=	inertial navigation system
MEAS	=	measurement transponder

The minimum and maximum values of the variables are listed in order to give the range of the variables for the purpose of scaling.

$A(1) = \text{DTFILF}$
 $A(2) = \text{DT2FIL}$
 $A(3) = A(1) * A(2)$
 $A(I) = -A(1)/\text{TAU}(I-9) \quad , \quad I = 10, 18$
 $B(1) = \text{SIGNO}(1) * \text{DLTT}$
 $B(2) = \text{SIGNO}(2) * \text{DLTT}$
 $B(I) = \text{SIGNO}(I) * (\text{UNID}/\text{TAU}(I-2))^{1/2} \quad I = 3, 9$
 $B(10) = \text{SIGNO}(10) * \text{DLTT}$
 $B(11) = \text{SIGNO}(11) * (\text{UNID}/\text{TAU}(11))^{1/2}$
 $B(12) = \text{SIGNO}(12) * (\text{UNID}/\text{TAU}(12))^{1/2}$
 $\text{SIG}(1) = 2. \quad \text{Position variance}$
 $\text{SIG}(2) = 10. \quad \text{Velocity variance}$
 $\text{SIG}(3) = .01 \quad \text{Gyro bias variance}$
 $\text{SIG}(4) = 1. \quad \text{Accelerometer bias variance}$
 $\text{SIG}(5) = 20. \quad \text{Wind velocity variance}$
 $\text{SIG}(6) = 1. \quad \text{Range bias variance}$
 $\text{SIG}(7) = 1. \quad \text{Bearing bias variance}$
 $\text{SIG}(8) = 1. \quad \text{Baro-altimeter bias variance}$
 $\text{UAR} = -1.$

4) Initial Variable Storage:

$\text{WT}(1, 1) = \text{SIG}(1)$
 $\text{WT}(2, 2) = \text{SIG}(1)$
 $\text{WT}(12, 12) = \text{SIG}(1)$
 $\text{WT}(3, 3) = \text{SIG}(2)$
 $\text{WT}(4, 4) = \text{SIG}(2)$
 $\text{WT}(13, 13) = \text{SIG}(2)$
 $\text{WT}(5, 5) = \text{SIG}(3)$
 $\text{WT}(6, 6) = \text{SIG}(3)$
 $\text{WT}(7, 7) = \text{SIG}(3)$

TABLE 1
FOREGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	THETA	INS	- 1.	1.
V	PSI	INS	- PI	PI
V	PHI	INS	- 1.	1.
P	PSIR	INITIAL	- PI	PI
V	GYRACC(1)	FG	- 1.	1.
V	GYRACC(2)	FG	- 1.	1.
V	GYRACC(3)	FG	- PI	PI
V	XHAT(1)	FG	- $4 \cdot 10^4$	$4 \cdot 10^4$
V	XHAT(2)	FG	- $4 \cdot 10^4$	$4 \cdot 10^4$
V	XHAT(3)	FG	- $4 \cdot 10^2$	$4 \cdot 10^2$
V	XHAT(4)	FG	- $4 \cdot 10^2$	$4 \cdot 10^2$
V	XHAT(5)	FG	- .1	.1
V	XHAT(6)	FG	- .1	.1
V	XHAT(7)	FG	- .1	.1
V	XHAT(8)	FG	0.	$1 \cdot 10^3$
V	XHAT(9)	FG	- .1	.1
V	XHAT(10)	FG	- 20.	20.
V	XHAT(11)	FG	- 20.	20.
V	XHAT(12)	FG	- $3 \cdot 10^4$	$3 \cdot 10^4$
V	XHAT(13)	FG	- $1 \cdot 10^2$	$1 \cdot 10^2$
V	XHAT(14)	FG	- .1	1.
V	XHAT(15)	FG	- $1 \cdot 10^2$	$1 \cdot 10^2$
S	COST	FG	- 1.	1.
S	SINT	FG	- 1.	1.
S	COSP	FG	- 1.	1.
S	SINP	FG	- 1.	1.
S	COSI	FG	- 1.	1.
S	SINI	FG	- 1.	1.

TABLE 1 (cont.)

FOREGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	AMXX	FG	- 1.	1.
S	AMXY	FG	- 1.	1.
S	AMXZ	FG	- 1.	1.
S	AMYX	FG	- 1.	1.
S	AMYY	FG	- 1.	1.
S	AMYZ	FG	- 1.	1.
S	AMZX	FG	- 1.	1.
S	AMZY	FG	- 1.	1.
S	AMZZ	FG	- 1.	1.
V	ACC(1)	INS	- 20.	20.
V	ACC(2)	INS	- 20.	20.
V	ACC(3)	INS	- 60.	60.
V	FXFYFZ(1)	FG	- 20.	20.
V	FXFYFZ(2)	FG	- 20.	20.
V	FXFYFZ(3)	FG	- 60.	60.
P	G	INITIAL	32.174	32.174
V	CACN(1)	FG	- 20.	20.
V	CACN(2)	FG	- 20.	20.
V	CACN(3)	FG	- 30.	30.
V	CACO(1)	FG	- 20.	20.
V	CACO(2)	FG	- 20.	20.
V	CACO(3)	FG	- 30.	30.
V	VELN(1)	FG	- $4 \cdot 10^2$	$4 \cdot 10^2$
V	VELN(2)	FG	- $4 \cdot 10^2$	$4 \cdot 10^2$
V	VELN(3)	FG	- $1 \cdot 10^2$	$1 \cdot 10^2$
V	VELO(1)	FG	- $4 \cdot 10^2$	$4 \cdot 10^2$
V	VELO(2)	FG	- $4 \cdot 10^2$	$4 \cdot 10^2$
V	VELO(3)	FG	- $1 \cdot 10^2$	$1 \cdot 10^2$

TABLE 1 (cont.)

FOREGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	POS(1)	FG	- 4.10^4	4.10^4
V	POS(2)	FG	- 4.10^4	4.10^4
V	POS(3)	FG	- 3.10^4	3.10^4
V	CS(1)	FG	- 35.	35.
V	CS(2)	FG	- 35.	35.
V	CS(3)	FG	- 35.	35.
V	CS(4)	FG	- 5.	5.
V	CS(5)	FG	- 5.	5.
V	CS(6)	FG	- 5.	5.
V	XS(1)	FG	- 4.10^4	4.10^4
V	XS(2)	FG	- 4.10^4	4.10^4
V	XS(3)	FG	- 3.10^4	3.10^4
V	XS(4)	FG	- 4.10^2	4.10^2
V	XS(5)	FG	- 4.10^2	4.10^2
V	XS(6)	FG	- 1.10^2	1.10^2
P	DT2FIL	INITIAL	.1	.1
I	ISTATE	BG, FG	0	1
P	DECAY	INITIAL	.9753	.9753
V	DX(1)	BG, FG	- 35.	35.
V	DX(2)	BG, FG	- 35.	35.
V	DX(3)	BG, FG	- 5.	5.
V	DX(4)	BG, FG	- 5.	5.
V	DX(5)	BG, FG	- .01	.01
V	DX(6)	BG, FG	- .01	.01
V	DX(7)	BG, FG	- .01	.01
V	DX(8)	BG, FG	0.0	0.0
V	DX(9)	BG, FG	0.0	0.0
V	DX(10)	BG, FG	- 5.	5.

TABLE 1 (cont.)

FOREGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	DX(11)	BG, FG	~ 5.	.5.
V	DX(12)	BG, FG	~ 60.	60.
V	DX(13)	BG, FG	~ 10.	10.
V	DX(14)	BG, FG	~ .5	.5
V	DX(15)	BG, FG	0.0	0.0
I	IFIL	FG, INITIAL	0	4
S	CSI	FG	~ 1.	1.
S	SSI	FG	~ 1.	1.
S	CTH ,	FG	~ 1.	1.
S	STH	FG	~ 1.	1.

The numbers of storage cells, by type, in the foreground loop are:

Variables	69
Scratch Variables	19
Permanent	4
Integer	2

TABLE 2

MIDGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	A(4)	MG	- 75.	75.
S	A(5)	MG	- 50.	50.
S	A(6)	MG	- 50.	50.
S	A(7)	MG	- 10.	10.
S	A(8)	MG	- 5.	5.
S	A(9)	MG	- 5.	5.
V	AP(1)	MG	.2	2.
V	AP(2)	MG	- 75.	75.
V	AP(3)	MG	- 50.	50.
V	AP(4)	MG	.2	2.
V	AP(5)	MG	- 75.	75.
V	AP(6)	MG	- 50.	50.
V	AP(7)	MG	- 75.	75.
V	AP(8)	MG	- 50.	50.
V	AP(9)	MG	- 75.	75.
V	AP(10)	MG	- 50.	50.
V	AP(11)	MG	- .02	- .002
V	AP(12)	MG	- .02	- .002
V	AP(13)	MG	- .02	- .002
V	AP(14)	MG	- .02	- .002
V	AP(15)	MG	- .02	- .002
V	AP(16)	MG	- .02	- .002
V	AP(17)	MG	- .02	- .002
V	AP(18)	MG	- 50.	50.
V	AP(19)	MG	- 50.	50.
V	AP(20)	MG	.2	2.
V	AP(21)	MG	.02	.2
V	AP(22)	MG	- 50.	50.

TABLE 2 (cont.)

MIDGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	AP(23)	MG	- 50.	50.
V	AP(24)	MG	.2	2.
V	AP(25)	MG	- .02	- .002
V	AP(26)	MG	- .02	- .002
S	XXN	MG	- 4.10^4	4.10^4
S	YYN	MG	- 4.10^4	4.10^4
S	ZZN	MG	- 3.10^4	3.10^4
S	XEN	MG	- 4.10^4	4.10^4
S	YEN	MG	- 4.10^4	4.10^4
S	ZEN	MG	- 3.10^4	3.10^4
S	SRAN	MG	1.10^7	1.10^{10}
V	XRANGE	MG	1.10^3	4.10^4
V	AZIMUT	MEAS	- .6	.6
V	AZIMUX	MG	- .6	.6
S	YRA	MG	- .3	.3
V	SRES(1)	MG	- .03	.03
S	SQRAN	MG	1.10^3	4.10^4
S	B	MG	- 1.10^{-7}	1.10^{-7}
S	UA(1)	MG	- 1.10^{-4}	1.10^{-4}
S	UA(2)	MG	- 1.10^{-3}	1.10^{-3}
S	UA(3)	MG	- 1.10^{-6}	1.10^{-6}
I	NRES(1)	MG	0	INTIME
P	TUMUCH(1)	INITIAL	.003	.003
V	HRES1(1)	MG	- 1.10^{-3}	1.10^{-3}
V	HRES1(2)	MG	- 1.10^{-2}	1.10^{-2}
V	HRES1(3)	MG	- 1.10^{-3}	1.10^{-3}
V	HRES1(4)	MG	- 1.10^{-2}	1.10^{-2}
V	HRES1(5)	MG	- 1.10^{-1}	1.10^{-1}

TABLE 2 (cont.)

MIDGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	HRES1(6)	MG	- 1.10^{-2}	1.10^{-2}
V	HRES1(7)	MG	- 1.10^{-2}	1.10^{-2}
V	HRES1(8)	MG	0.0	0.0
V	HRES1(9)	MG	0.0	0.0
V	HRES1(10)	MG	0.0	0.0
V	HRES1(11)	MG	0.0	0.0
V	HRES1(12)	MG	- 1.10^{-4}	1.10^{-4}
V	HRES1(13)	MG	- 1.10^{-4}	1.10^{-4}
V	HRES1(14)	MG	- 1.10^{-5}	1.10^{-5}
V	HRES1(15)	MG	0.0	0.0
S	XYN	MG	1.10^7	1.10^9
S	REV	MG	1.10^7	1.10^9
S	SXYN	MG	- 1.10^{-3}	1.10^{-3}
S	ZRET	MG	- 1.10^{-7}	1.10^{-7}
V	ELEVAT	MEAS	- .5	.5
V	ELEVAX	MG	- .5	.5
S	YRE	MG	- .5	.5
V	SRES(2)	MG (ELEVATION)	- .03	.03
S	UE(1)	MG	- 1.10^{-4}	1.10^{-4}
S	UE(2)	MG	- 1.10^{-4}	1.10^{-4}
S	UE(3)	MG	- 1.10^{-3}	1.10^{-3}
V	HRES2(1)	MG	- 1.10^{-3}	1.10^{-3}
V	HRES2(2)	MG	- 1.10^{-4}	1.10^{-4}
V	HRES2(3)	MG	- 1.10^{-3}	1.10^{-3}
V	HRES2(4)	MG	- 1.10^{-3}	1.10^{-3}
V	HRES2(5)	MG	- 1.10^{-2}	1.10^{-2}
V	HRES2(6)	MG	- 1.10^{-2}	1.10^{-2}
V	HRES2(7)	MG	- 1.10^{-3}	1.10^{-3}

TABLE 2 (cont.)

MIDGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	HRES2(8)	MG	0.0	0.0
V	HRES2(9)	MG	0.0	0.0
V	HRES2(10)	MG	0.0	0.0
V	HRES2(11)	MG	0.0	0.0
V	HRES2(12)	MG	- 1.10 ⁻²	1.10 ⁻²
V	HRES2(13)	MG	- 1.10 ⁻²	1.10 ⁻²
V	HRES2(14)	MG	- 1.10 ⁻³	1.10 ⁻³
V	HRES2(15)	MG	0.0	0.0
L	VALID(1)	MEAS	FALSE	TRUE
L	VALID(2)	MEAS	FALSE	TRUE
L	VALID(3)	MEAS	FALSE	TRUE
V	ERANGE	MG	1.10 ³	4.10 ⁴
V	RANGE	MEAS	1.10 ³	4.10 ⁴
S	YRR	MG	- 1.10 ⁴	1.10 ⁴
V	SRES(3)	MG	-1500.	1500.
P	TUMUCH(3)	INITIAL	150.	150.
S	UR(1)	MG	- 1.	1.
S	UR(2)	MG	- 1.	1.
S	UR(3)	MG	- 1.	1.
V	HRES3(1)	MG	- 10.	10.
V	HRES3(2)	MG	- 10.	10.
V	HRES3(3)	MG	- 15.	15.
V	HRES3(4)	MG	- 15.	15.
V	HRES3(5)	MG	- 5.10 ²	5.10 ²
V	HRES3(6)	MG	- 5.10 ²	5.10 ²
V	HRES3(7)	MG	- 5.10 ²	5.10 ²
V	HRES3(8)	MG	0.0	0.0
V	HRES3(9)	MG	0.0	0.0

TABLE 2 (cont.)

MIDGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	HRES3(10)	MG	0.0	0.0
V	HRES3(11)	MG	0.0	0.0
V	HRES3(12)	MG	- 1.	1.
V	HRES3(13)	MG	- 1.	1.
V	HRES3(14)	MG	- .01	.01
V	HRES3(15)	MG	0.0	0.0
I	NRES(2)	MG	0	INTIME
I	NRES(3)	MG	0	INTIME
L	RADAR	MG	FALSE	TRUE
V	ALT	MG	0.	150.
V	ALTU	MEAS	0.	150.
V	POS(3)	FG	- 1.10^4	1.10^4
S	YRAL	MG	- 1.10^2	1.10^2
P	UAR	INITIAL	- 1.	- 1.
V	SRES(2) (RADAR)	MG	-200.	200.
V	HRES2(1)	MG	0.0	0.0
V	HRES2(2)	MG	0.0	0.0
V	HRES2(3)	MG	0.0	0.0
V	HRES2(4)	MG	-0.0	0.0
V	HRES2(5)	MG	-100.	100.
V	HRES2(6)	MG	-100.	100.
V	HRES2(7)	MG	0.0	0.0
V	HRES2(8)	MG	0.0	0.0
V	HRES2(9)	MG	0.0	0.0
V	HRES2(10)	MG	0.0	0.0
V	HRES2(11)	MG	0.0	0.0
V	HRES2(12)	MG	- 15.	15.
V	HRES2(13)	MG	- 15.	15.

TABLE 2 (cont.)

MIDGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	HRES2(14)	MG	- 2.	2.
V	HRES2(15)	MG	0.0	0.0
V	AIRSP	MEAS	0.	4.10^2
V	AIR	MG	0.	4.10^2
S	VX	MG	-400.	400.
S	VY	MG	-400.	400.
S	VZ	MG	-100.	100.
S	YRAR	MG	- 20.	20.
P	TUMUCH(5)	INITIAL	- 20.	20.
V	SRES(4)	MG	-200.	200.
I	NRES(4)	MG	0	INTIME
S	UAI(1)	MG	- 1.	1.
S	UAI(2)	MG	- 1.	1.
V	HRES4(1)	MG	0.0	0.0
V	HRES4(2)	MG	0.0	0.0
V	HRES4(3)	MG	0.0	0.0
V	HRES4(4)	MG	0.0	0.0
V	HRES4(5)	MG	0.0	0.0
V	HRES4(6)	MG	0.0	0.0
V	HRES4(7)	MG	0.0	0.0
V	HRES4(8)	MG	0.0	0.0
V	HRES4(9)	MG	0.0	0.0
V	HRES4(10)	MG	- .2	.2
V	HRES4(11)	MG	- .2	.2
V	HRES4(12)	MG	0.0	0.0
V	HRES4(13)	MG	0.0	0.0
V	HRES4(14)	MG	0.0	0.0
V	HRES4(15)	MG	0.0	0.0

TABLE 2 (cont.)

MIDGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	DX(1)	BG, MG	- 35.	35.
V	DX(2)	BG, MG	- 35.	35.
V	DX(3)	BG, MG	- 5.	5.
V	DX(4)	BG, MG	- 5.	5.
V	DX(5)	BG, MG	- .01	.01
V	DX(6)	BG, MG	- .01	.01
V	DX(7)	BG, MG	- .01	.01
V	DX(8)	BG, MG	0.0	0.0
V	DX(9)	BG, MG	0.0	0.0
V	DX(10)	BG, MG	- 5.	5.
V	DX(11)	BG, MG	- 5.	5.
V	DX(12)	BG, MG	- 60.	60.
V	DX(13)	BG, MG	- 10.	10.
V	DX(14)	BG, MG	- .5	.5
V	DX(15)	BG, MG	0.0	0.0
S	S(1)	MG	- 35.	35.
S	S(2)	MG	- 35.	35.
S	S(3)	MG	- 5.	5.
S	S(4)	MG	- 5.	5.
S	S(5)	MG	- .01	.01
S	S(6)	MG	- .01	.01
S	S(7)	MG	- .01	.01
S	S(8)	MG	0.0	0.0
S	S(9)	MG	0.0	0.0
S	S(10)	MG	- 5.	5.
S	S(11)	MG	- 5.	5.
S	S(12)	MG	- 60.	60.
S	S(13)	MG	- 10.	10.

TABLE 2 (cont.)

MIDGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	S(14)	MG	- 5.	5.
S	S(15)	MG	0.0	0.0
P	NE(1)	INITIAL	1	1
P	NE(2)	INITIAL	4	4
P	NE(3)	INITIAL	7	7
P	NE(4)	INITIAL	9	9
P	NE(5)	INITIAL	11	11
P	NE(6)	INITIAL	12	12
P	NE(7)	INITIAL	13	13
P	NE(8)	INITIAL	14	14
P	NE(9)	INITIAL	15	15
P	NE(10)	INITIAL	16	16
P	NE(11)	INITIAL	17	17
P	NE(12)	INITIAL	18	18
P	NE(13)	INITIAL	22	22
P	NE(14)	INITIAL	25	25
P	NE(15)	INITIAL	26	26
P	NE(16)	INITIAL	27	27
P	NR(1)	INITIAL	3	3
P	NR(2)	INITIAL	6	6
P	NR(3)	INITIAL	7	7
P	NR(4)	INITIAL	4	4
P	NR(5)	INITIAL	5	5
P	NR(6)	INITIAL	7	7
P	NR(7)	INITIAL	6	6
P	NR(8)	INITIAL	7	7
P	NR(9)	INITIAL	5	5
P	NR(10)	INITIAL	7	7

TABLE 2 (cont.)

MIDGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
P	NR(11)	INITIAL	5	5
P	NR(12)	INITIAL	6	6
P	NR(13)	INITIAL	7	7
P	NR(14)	INITIAL	8	8
P	NR(15)	INITIAL	9	9
P	NR(16)	INITIAL	10	10
P	NR(17)	INITIAL	11	11
P	NR(18)	INITIAL	5	5
P	NR(19)	INITIAL	6	6
P	NR(20)	INITIAL	13	13
P	NR(21)	INITIAL	14	14
P	NR(22)	INITIAL	5	5
P	NR(23)	INITIAL	6	6
P	NR(24)	INITIAL	14	14
P	NR(25)	INITIAL	14	14
P	NR(26)	INITIAL	15	15
I	NRES(4)	MG	0	INTIME
I	MPROC	BG, MG	0	1
I	ISTATE	FG, MG	0	1
I	MESCNT	BG, MG	0	1
P	TUMUCH(2) (ELEVATION)	INITIAL	.003	.003
P	TUMUCH(2) (RADAR)	INITIAL	1.10^5	1.10^5

The following storage cells are required for transfer and save functions for values already computed in the midground loop:

<u>Type</u>	<u>Name</u>	<u>Previous Name</u>	<u>Number of Cells</u>
I	IND(J)	NRES(J)	J = 1, 4
V	YR(J)	SRES(J)	J = 1, 4
V	H1(J)	HRES1(J)	J = 1, 15
V	H2(J)	HRES2(J)	J = 1, 15
V	H3(J)	HRES3(J)	J = 1, 15
V	H4(J)	HRES4(J)	J = 1, 15
V	APS(J)	AP(J)	J = 1, 26

The total number of cells required for the midground computation, by type, are given below:

Variables	223
Scratch Variables	53
Permanent	48
Integer	12
Logical	4

TABLE 3
BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	D(1)	BG	-150.	150.
S	D(2)	BG	-150.	150.
S	D(3)	BG	- 50.	50.
S	D(4)	BG	- 50.	50.
S	D(5)	BG	- 5.	5.
S	D(6)	BG	- 5.	5.
S	D(7)	BG	- 5.	5.
S	D(8)	BG	0.0	0.0
S	D(9)	BG	0.0	0.0
S	D(10)	BG	- 5.	5.
S	D(11)	BG	- 5.	5.
S	D(12)	BG	-150.	150.
S	D(13)	BG	- 50.	50.
S	D(14)	BG	- .1	.1
S	D(15)	BG	0.0	0.0
S	VAR(1)	BG	1.10^{-4}	1.10^{-3}
S	VAR(2)	BG	1.10^{-3}	1.10^{-2}
	(ELEVATION)			
S	VAR(3)	BG	1.10^4	1.10^5
S	VAR(4)	BG	400.	4000.
S	VAR(2) (RADAR)	BG	90.	1000.
S	E(1)	BG	- 30.	30.
S	E(2)	BG	- 30.	30.
	(ELEVATION)			
S	E(3)	BG	- 1.10^{-2}	1.10^{-2}
S	E(4)	BG	- .1	.1
S	E(2) (RADAR)	BG	- 20.	20.
S	AD(1)	BG	1.10^{-4}	1.10^{-2}
S	AD(2)	BG	1.10^{-4}	1.10^{-2}
	(ELEVATION)			

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	AD(3)	BG	1.10^5	1.10^6
S	AD(4)	BG	1.10^3	1.10^4
S	AD(2) (RADAR)	BG	1.10^3	1.10^4
S	BD(1)	BG	- 1.10^4	1.10^4
S	BD(2)	BG	- 1.10^4	1.10^4
S	BD(3)	BG	- 1.10^3	1.10^3
S	BD(4)	BG	- 1.10^3	1.10^3
S	BD(5)	BG	- 1.0	1.0
S	BD(6)	BG	- 1.0	1.0
S	BD(7)	BG	- 1.0	1.0
S	BD(8)	BG	0.0	0.0
S	BD(9)	BG	0.0	0.0
S	BD(10)	BG	-100.	100.
S	BD(11)	BG	-100.	100.
S	BD(12)	BG	- 1.10^4	1.10^4
S	BD(13)	BG	- 1.10^3	1.10^3
S	BD(14)	BG	- 12.	12.
S	BD(15)	BG	0.0	0.0
V	DX(1)	BG	- 35.	35.
V	DX(2)	BG	- 35.	35.
V	DX(3)	BG	- 5.	5.
V	DX(4)	BG	- 5.	5.
V	DX(5)	BG	- .01	.01
V	DX(6)	BG	- .01	.01
V	DX(7)	BG	- .01	.01
V	DX(8)	BG	0.0	0.0
V	DX(9)	BG	0.0	0.0
V	DX(10)	BG	- 5.	5.

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	DX(11)	BG	- 5.	5.
V	DX(12)	BG	- 60.	60.
V	DX(13)	BG	- 10.	10.
V	DX(14)	BG	- 5.	5.
V	DX(15)	BG	0.0	0.0
V	WT(1, 1)	BG	- 1.10^2	1.10^2
V	WT(2, 1)	BG	- 50.	50.
V	WT(3, 1)	BG	- 10.	10.
V	WT(4, 1)	BG	- 10.	10.
V	WT(5, 1)	BG	- 1.	1.
V	WT(6, 1)	BG	- 1.	1.
V	WT(7, 1)	BG	- 1.	1.
V	WT(8, 1)	BG	0.0	0.0
V	WT(9, 1)	BG	0.0	0.0
V	WT(10, 1)	BG	0.0	0.0
V	WT(11, 1)	BG	0.0	0.0
V	WT(12, 1)	BG	- 10.	10.
V	WT(13, 1)	BG	- 1.	1.
V	WT(14, 1)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(15, 1)	BG	0.0	0.0
V	WT(1, 2)	BG	- 50.	50.
V	WT(2, 2)	BG	-100.	100.
V	WT(3, 2)	BG	- 10.	10.
V	WT(4, 2)	BG	- 10.	10.
V	WT(5, 2)	BG	- 1.	1.
V	WT(6, 2)	BG	- 1.	1.
V	WT(7, 2)	BG	- 1.	1.
V	WT(8, 2)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	WT(9, 2)	BG	0.0	0.0
V	WT(10, 2)	BG	0.0	0.0
V	WT(11, 2)	BG	0.0	0.0
V	WT(12, 2)	BG	- 10.	10.
V	WT(13, 2)	BG	- 1.	1.
V	WT(14, 2)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(15, 2)	BG	0.0	0.0
V	WT(1, 3)	BG	- 5.	5.
V	WT(2, 3)	BG	- 5.	5.
V	WT(3, 3)	BG	- 5.	5.
V	WT(4, 3)	BG	- 1.	1.
V	WT(5, 3)	BG	- 1.	1.
V	WT(6, 3)	BG	- 1.	1.
V	WT(7, 3)	BG	- 1.	1.
V	WT(8, 3)	BG	0.0	0.0
V	WT(9, 3)	BG	0.0	0.0
V	WT(10, 3)	BG	0.0	0.0
V	WT(11, 3)	BG	0.0	0.0
V	WT(12, 3)	BG	- 1.	1.
V	WT(13, 3)	BG	- .1	.1
V	WT(14, 3)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(15, 3)	BG	0.0	0.0
V	WT(1, 4)	BG	- 5.	5.
V	WT(2, 4)	BG	- 5.	5.
V	WT(3, 4)	BG	- 5.	5.
V	WT(4, 4)	BG	- 5.	5.
V	WT(5, 4)	BG	- 1.	1.
V	WT(6, 4)	BG	- 1.	1.

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	WT(7, 4)	BG	- 1.	1.
V	WT(8, 4)	BG	0.0	0.0
V	WT(9, 4)	BG	0.0	0.0
V	WT(10, 4)	BG	0.0	0.0
V	WT(11, 4)	BG	0.0	0.0
V	WT(12, 4)	BG	- 1.	1.
V	WT(13, 4)	BG	- 1.	1.
V	WT(14, 4)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(15, 4)	BG	0.0	0.0
V	WT(1, 5)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(2, 5)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(3, 5)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(4, 5)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(5, 5)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(6, 5)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(7, 5)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(8, 5)	BG	0.0	0.0
V	WT(9, 5)	BG	0.0	0.0
V	WT(10, 5)	BG	0.0	0.0
V	WT(11, 5)	BG	0.0	0.0
V	WT(12, 5)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(13, 5)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(14, 5)	BG	- 1.10^{-6}	1.10^{-6}
V	WT(15, 5)	BG	0.0	0.0
V	WT(1, 6)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(2, 6)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(3, 6)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(4, 6)	BG	- 1.10^{-2}	1.10^{-2}

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	WT(5, 6)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(6, 6)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(7, 6)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(8, 6)	BG	0.0	0.0
V	WT(9, 6)	BG	0.0	0.0
V	WT(10, 6)	BG	0.0	0.0
V	WT(11, 6)	BG	0.0	0.0
V	WT(12, 6)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(13, 6)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(14, 6)	BG	- 1.10^{-6}	1.10^{-6}
V	WT(15, 6)	BG	0.0	0.0
V	WT(1, 7)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(2, 7)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(3, 7)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(4, 7)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(5, 7)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(6, 7)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(7, 7)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(8, 7)	BG	0.0	0.0
V	WT(9, 7)	BG	0.0	0.0
V	WT(10, 7)	BG	0.0	0.0
V	WT(11, 7)	BG	0.0	0.0
V	WT(12, 7)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(13, 7)	BG	- 1.10^{-3}	1.10^{-3}
V	WT(14, 7)	BG	- 1.10^{-6}	1.10^{-6}
V	WT(15, 7)	BG	0.0	0.0
V	WT(1, 8)	BG	0.0	0.0
V	WT(2, 8)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	WT(3, 8)	BG	0.0	0.0
V	WT(4, 8)	BG	0.0	0.0
V	WT(5, 8)	BG	0.0	0.0
V	WT(6, 8)	BG	0.0	0.0
V	WT(7, 8)	BG	0.0	0.0
V	WT(8, 8)	BG	-500.	500.
V	WT(9, 8)	BG	0.0	0.0
V	WT(10, 8)	BG	0.0	0.0
V	WT(11, 8)	BG	0.0	0.0
V	WT(12, 8)	BG	0.0	0.0
V	WT(13, 8)	BG	0.0	0.0
V	WT(14, 8)	BG	0.0	0.0
V	WT(15, 8)	BG	0.0	0.0
V	WT(1, 9)	BG	0.0	0.0
V	WT(2, 9)	BG	0.0	0.0
V	WT(3, 9)	BG	0.0	0.0
V	WT(4, 9)	BG	0.0	0.0
V	WT(5, 9)	BG	0.0	0.0
V	WT(6, 9)	BG	0.0	0.0
V	WT(7, 9)	BG	0.0	0.0
V	WT(8, 9)	BG	0.0	0.0
V	WT(9, 9)	BG	- .1	.1
V	WT(10, 9)	BG	0.0	0.0
V	WT(11, 9)	BG	0.0	0.0
V	WT(12, 9)	BG	0.0	0.0
V	WT(13, 9)	BG	0.0	0.0
V	WT(14, 9)	BG	0.0	0.0
V	WT(15, 9)	BG	0.0	0.0

TABLE- 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	WT(1, 10)	BG	0.0	0.0
V	WT(2, 10)	BG	0.0	0.0
V	WT(3, 10)	BG	0.0	0.0
V	WT(4, 10)	BG	0.0	0.0
V	WT(5, 10)	BG	0.0	0.0
V	WT(6, 10)	BG	0.0	0.0
V	WT(7, 10)	BG	0.0	0.0
V	WT(8, 10)	BG	0.0	0.0
V	WT(9, 10)	BG	0.0	0.0
V	WT(10, 10)	BG	- 20.	20.
V	WT(11, 10)	BG	- 1.	1.
V	WT(12, 10)	BG	0.0	0.0
V	WT(13, 10)	BG	0.0	0.0
V	WT(14, 10)	BG	0.0	0.0
V	WT(15, 10)	BG	0.0	0.0
V	WT(1, 11)	BG	0.0	0.0
V	WT(2, 11)	BG	0.0	0.0
V	WT(3, 11)	BG	0.0	0.0
V	WT(4, 11)	BG	0.0	0.0
V	WT(5, 11)	BG	0.0	0.0
V	WT(6, 11)	BG	0.0	0.0
V	WT(7, 11)	BG	0.0	0.0
V	WT(8, 11)	BG	0.0	0.0
V	WT(9, 11)	BG	0.0	0.0
V	WT(10, 11)	BG	- 1.	1.
V	WT(11, 11)	BG	- 20.	20.
V	WT(12, 11)	BG	0.0	0.0
V	WT(13, 11)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	WT(14, 11)	BG	0.0	0.0
V	WT(15, 11)	BG	0.0	0.0
V	WT(1, 12)	BG	-100.	100.
V	WT(2, 12)	BG	-100.	100.
V	WT(3, 12)	BG	- 50.	50.
V	WT(4, 12)	BG	- 50.	50.
V	WT(5, 12)	BG	- 1.	1.
V	WT(6, 12)	BG	- 1.	1.
V	WT(7, 12)	BG	- 1.	1.
V	WT(8, 12)	BG	0.0	0.0
V	WT(9, 12)	BG	0.0	0.0
V	WT(10, 12)	BG	0.0	0.0
V	WT(11, 12)	BG	0.0	0.0
V	WT(12, 12)	BG	-500.	500.
V	WT(13, 12)	BG	- 50.	50.
V	WT(14, 12)	BG	- 1.	1.
V	WT(15, 12)	BG	0.0	0.0
V	WT(1, 13)	BG	- 50.	50.
V	WT(2, 13)	BG	- 50.	50.
V	WT(3, 13)	BG	- 50.	50.
V	WT(4, 13)	BG	- 50.	50.
V	WT(5, 13)	BG	- 1.	1.
V	WT(6, 13)	BG	- 1.	1.
V	WT(7, 13)	BG	- 1.	1.
V	WT(8, 13)	BG	0.0	0.0
V	WT(9, 13)	BG	0.0	0.0
V	WT(10, 13)	BG	0.0	0.0
V	WT(11, 13)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	WT(12,13)	BG	- 50.	50.
V	WT(13,13)	BG	- 10.	10.
V	WT(14,13)	BG	- 5.	5.
V	WT(15,13)	BG	0.0	0.0
V	WT(1,14)	BG	- 1.	1.
V	WT(2,14)	BG	- 1.	1.
V	WT(3,14)	BG	- 1.	1.
V	WT(4,14)	BG	- 1.	1.
V	WT(5,14)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(6,14)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(7,14)	BG	- 1.10^{-2}	1.10^{-2}
V	WT(8,14)	BG	0.0	0.0
V	WT(9,14)	BG	0.0	0.0
V	WT(10,14)	BG	0.0	0.0
V	WT(11,14)	BG	0.0	0.0
V	WT(12,14)	BG	- 1.	1.
V	WT(13,14)	BG	- 1.	1.
V	WT(14,14)	BG	- 1.	1.
V	WT(15,14)	BG	0.0	0.0
V	WT(1,15)	BG	0.0	0.0
V	WT(2,15)	BG	0.0	0.0
V	WT(3,15)	BG	0.0	0.0
V	WT(4,15)	BG	0.0	0.0
V	WT(5,15)	BG	0.0	0.0
V	WT(6,15)	BG	0.0	0.0
V	WT(7,15)	BG	0.0	0.0
V	WT(8,15)	BG	0.0	0.0
V	WT(9,15)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	WT(10, 15)	BG	0.0	0.0
V	WT(11, 15)	BG	0.0	0.0
V	WT(12, 15)	BG	0.0	0.0
V	WT(13, 15)	BG	0.0	0.0
V	WT(14, 15)	BG	0.0	0.0
V	WT(15, 15)	BG	-200.	200.
S	SS(1)	BG	- 35.	35.
S	SS(2)	BG	- 35.	35.
S	SS(3)	BG	- 5.	5.
S	SS(4)	BG	- 5.	5.
S	SS(5)	BG	- .01	.01
S	SS(6)	BG	- .01	.01
S	SS(7)	BG	- .01	.01
S	SS(8)	BG	0.0	0.0
S	SS(9)	BG	0.0	0.0
S	SS(10)	BG	- 5.	5.
S	SS(11)	BG	- 5.	5.
S	SS(12)	BG	- 60.	60.
S	SS(13)	BG	- 10.	10.
S	SS(14)	BG	- 5.	5.
S	SS(15)	BG	0.0	0.0
V	WT(16, 3)	BG	- .5	.5
V	WT(17, 4)	BG	- .5	.5
V	WT(18, 5)	BG	- .002	.002
V	WT(19, 6)	BG	- .002	.002
V	WT(20, 7)	BG	- .002	.002
V	WT(21, 8)	BG	-100.	100.
V	WT(22, 9)	BG	- .006	.006

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
V	WT(23,10)	BG	- 4.	4.
V	WT(24,11)	BG	- 4.	4.
V	WT(25,13)	BG	- .5	.5
V	WT(26,14)	BG	- .002	.002
V	WT(27,15)	BG	- 40.	40.
S	DD(1)	BG	- 50.	50.
S	DD(2)	BG	- 5.	5.
S	DD(3)	BG	- 5.	5.
S	DD(4)	BG	- 5.	5.
S	DD(5)	BG	- 1.	1.
S	DD(6)	BG	- 1.	1.
S	DD(7)	BG	- 1.	1.
S	DD(8)	BG	0.0	0.0
S	DD(9)	BG	0.0	0.0
S	DD(10)	BG	0.0	0.0
S	DD(11)	BG	0.0	0.0
S	DD(12)	BG	- 5.	5.
S	DD(13)	BG	- 1.	1.
S	DD(14)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(15)	BG	0.0	0.0
S	DD(16)	BG	0.0	0.0
S	DD(17)	BG	0.0	0.0
S	DD(18)	BG	0.0	0.0
S	DD(19)	BG	0.0	0.0
S	DD(20)	BG	0.0	0.0
S	DD(21)	BG	0.0	0.0
S	DD(22)	BG	0.0	0.0
S	DD(23)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	DD(24)	BG	0.0	0.0
S	DD(25)	BG	0.0	0.0
S	DD(26)	BG	0.0	0.0
S	DD(27)	BG	0.0	0.0
S	WS(2)	BG	- 1.	1.
S	WS(3)	BG	- 1.	1.
S	WS(4)	BG	- $1 \cdot 10^{-1}$	$1 \cdot 10^{-1}$
S	WS(5)	BG	- $1 \cdot 10^{-3}$	$1 \cdot 10^{-3}$
S	WS(6)	BG	- $1 \cdot 10^{-3}$	$1 \cdot 10^{-3}$
S	WS(7)	BG	- $1 \cdot 10^{-3}$	$1 \cdot 10^{-3}$
S	WS(8)	BG	0.0	0.0
S	WS(9)	BG	0.0	0.0
S	WS(10)	BG	0.0	0.0
S	WS(11)	BG	0.0	0.0
S	WS(12)	BG	- 10.	10.
S	WS(13)	BG	- 1.	1.
S	WS(14)	BG	- $1 \cdot 10^{-1}$	$1 \cdot 10^{-1}$
S	WS(15)	BG	0.0	0.0
S	BETU(1)	BG	- $1 \cdot 10^{-2}$	$1 \cdot 10^{-2}$
S	DSIG(1)	BG	- $1 \cdot 10^2$	$1 \cdot 10^2$
S	DD(2)	BG	- 50.	50.
S	DD(3)	BG	- 5.	5.
S	DD(4)	BG	- 5.	5.
S	DD(5)	BG	- 1.	1.
S	DD(6)	BG	- 1.	1.
S	DD(7)	BG	- 1.	1.
S	DD(8)	BG	0.0	0.0
S	DD(9)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	DD(10)	BG	0.0	0.0
S	DD(11)	BG	0.0	0.0
S	DD(12)	BG	- 5.	5.
S	DD(13)	BG	- 1.	1.
S	DD(14)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(15)	BG	0.0	0.0
S	DD(16)	BG	0.0	0.0
S	DD(17)	BG	0.0	0.0
S	DD(18)	BG	0.0	0.0
S	DD(19)	BG	0.0	0.0
S	DD(20)	BG	0.0	0.0
S	DD(21)	BG	0.0	0.0
S	DD(22)	BG	0.0	0.0
S	DD(23)	BG	0.0	0.0
S	DD(24)	BG	0.0	0.0
S	DD(25)	BG	0.0	0.0
S	DD(26)	BG	0.0	0.0
S	DD(27)	BG	0.0	0.0
S	WS(3)	BG	- .1	.1
S	WS(4)	BG	- 1.	1.
S	WS(5)	BG	- 1.10^{-3}	1.10^{-3}
S	WS(6)	BG	- 1.10^{-3}	1.10^{-3}
S	WS(7)	BG	- 1.10^{-3}	1.10^{-3}
S	WS(8)	BG	0.0	0.0
S	WS(9)	BG	0.0	0.0
S	WS(10)	BG	0.0	0.0
S	WS(11)	BG	0.0	0.0
S	WS(12)	BG	- 5.	5.

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	WS(13)	BG	- 1.	1.
S	WS(14)	BG	- 1.10^{-2}	1.10^{-2}
S	WS(15)	BG	0.0	0.0
S	BETU(2)	BG	- 1.10^{-1}	1.10^{-1}
S	DSIG(2)	BG	-100.	100.
S	DD(3)	BG	- 5.	5.
S	DD(4)	BG	- 1.	1.
S	DD(5)	BG	- 1.	1.
S	DD(6)	BG	- 1.	1.
S	DD(7)	BG	- 1.	1.
S	DD(8)	BG	0.0	0.0
S	DD(9)	BG	0.0	0.0
S	DD(10)	BG	0.0	0.0
S	DD(11)	BG	0.0	0.0
S	DD(12)	BG	- 5.	5.
S	DD(13)	BG	- 1.	1.
S	DD(14)	BG	- 1.10^{-3}	1.10^{-3}
S	DD(15)	BG	0.0	0.0
S	DD(16)	BG	0.0	0.0
S	DD(17)	BG	- .5	.5
S	DD(18)	BG	0.0	0.0
S	DD(19)	BG	0.0	0.0
S	DD(20)	BG	0.0	0.0
S	DD(21)	BG	0.0	0.0
S	DD(22)	BG	0.0	0.0
S	DD(23)	BG	0.0	0.0
S	DD(24)	BG	0.0	0.0
S	DD(25)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	DD(26)	BG	0.0	0.0
S	DD(27)	BG	0.0	0.0
S	WS(4)	BG	- 1.	1.
S	WS(5)	BG	- 1.10^{-2}	1.10^{-2}
S	WS(6)	BG	- 1.10^{-2}	1.10^{-2}
S	WS(7)	BG	- 1.10^{-2}	1.10^{-2}
S	WS(8)	BG	0.0	0.0
S	WS(9)	BG	0.0	0.0
S	WS(10)	BG	0.0	0.0
S	WS(11)	BG	0.0	0.0
S	WS(12)	BG	- 50.	50.
S	WS(13)	BG	- 5.	5.
S	WS(14)	BG	- 1.	1.
S	WS(15)	BG	0.0	0.0
S	BETU(3)	BG	- 1.	1.
S	DSIG(3)	BG	- 5.	5.
S	DD(4)	BG	- 5.	5.
S	DD(5)	BG	- 1.	1.
S	DD(6)	BG	- 1.	1.
S	DD(7)	BG	- 1.	1.
S	DD(8)	BG	0.0	0.0
S	DD(9)	BG	0.0	0.0
S	DD(10)	BG	0.0	0.0
S	DD(11)	BG	0.0	0.0
S	DD(12)	BG	- 1.	1.
S	DD(13)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(14)	BG	- 1.10^{-4}	1.10^{-4}
S	DD(15)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	DD(16)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(17)	BG	- 5.	5.
S	DD(18)	BG	0.0	0.0
S	DD(19)	BG	0.0	0.0
S	DD(20)	BG	0.0	0.0
S	DD(21)	BG	0.0	0.0
S	DD(22)	BG	0.0	0.0
S	DD(23)	BG	0.0	0.0
S	DD(24)	BG	0.0	0.0
S	DD(25)	BG	0.0	0.0
S	DD(26)	BG	0.0	0.0
S	DD(27)	BG	0.0	0.0
S	WS(5)	BG	- 1.10^{-2}	1.10^{-2}
S	WS(6)	BG	- 1.10^{-2}	1.10^{-2}
S	WS(7)	BG	- 1.10^{-2}	1.10^{-2}
S	WS(8)	BG	0.0	0.0
S	WS(9)	BG	0.0	0.0
S	WS(10)	BG	0.0	0.0
S	WS(11)	BG	0.0	0.0
S	WS(12)	BG	- 20.	20.
S	WS(13)	BG	- 5.	5.
S	WS(14)	BG	- .1	.1
S	WS(15)	BG	0.0	0.0
S	BETU(4)	BG	- 1.	1.
S	DSIG(4)	BG	- 1.	1.
S	DD(5)	BG	- 1.10^{-1}	1.10^{-1}
S	DD(6)	BG	- 1.10^{-3}	1.10^{-3}
S	DD(7)	BG	- 1.10^{-3}	1.10^{-3}
S	DD(8)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	DD(9)	BG	0.0	0.0
S	DD(10)	BG	0.0	0.0
S	DD(11)	BG	0.0	0.0
S	DD(12)	BG	- 1.10^{-4}	1.10^{-4}
S	DD(13)	BG	- 1.10^{-4}	1.10^{-4}
S	DD(14)	BG	- 1.10^{-6}	1.10^{-6}
S	DD(15)	BG	0.0	0.0
S	DD(16)	BG	- 1.10^{-3}	1.10^{-3}
S	DD(17)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(18)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(19)	BG	0.0	0.0
S	DD(20)	BG	0.0	0.0
S	DD(21)	BG	0.0	0.0
S	DD(22)	BG	0.0	0.0
S	DD(23)	BG	0.0	0.0
S	DD(24)	BG	0.0	0.0
S	DD(25)	BG	0.0	0.0
S	DD(26)	BG	0.0	0.0
S	DD(27)	BG	0.0	0.0
S	WS(6)	BG	- 1.10^{-1}	1.10^{-1}
S	WS(7)	BG	- 1.	1.
S	WS(8)	BG	0.0	0.0
S	WS(9)	BG	0.0	0.0
S	WS(10)	BG	0.0	0.0
S	WS(11)	BG	0.0	0.0
S	WS(12)	BG	- 1.10^2	1.10^2
S	WS(13)	BG	- 50.	50.
S	WS(14)	BG	- 5.	5.

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	WS(15)	BG	0.0	0.0
S	BETU(5)	BG	- 1.10^5	1.10^5
S	DSIG(5)	BG	-100.	100.
S	DD(6)	BG	- 1.10^{-1}	1.10^{-1}
S	DD(7)	BG	- 1.10^{-3}	1.10^{-3}
S	DD(8)	BG	0.0	0.0
S	DD(9)	BG	0.0	0.0
S	DD(10)	BG	0.0	0.0
S	DD(11)	BG	0.0	0.0
S	DD(12)	BG	- 1.10^{-3}	1.10^{-3}
S	DD(13)	BG	- 1.10^{-3}	1.10^{-3}
S	DD(14)	BG	- 1.10^{-6}	1.10^{-6}
S	DD(15)	BG	0.0	0.0
S	DD(16)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(17)	BG	- 1.10^{-3}	1.10^{-3}
S	DD(18)	BG	- 1.10^{-3}	1.10^{-3}
S	DD(19)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(20)	BG	0.0	0.0
S	DD(21)	BG	0.0	0.0
S	DD(22)	BG	0.0	0.0
S	DD(23)	BG	0.0	0.0
S	DD(24)	BG	0.0	0.0
S	DD(25)	BG	0.0	0.0
S	DD(26)	BG	0.0	0.0
S	DD(27)	BG	0.0	0.0
S	WS(7)	BG	- 1.	1.
S	WS(8)	BG	0.0	0.0
S	WS(9)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	WS(10)	BG	0.0	0.0
S	WS(11)	BG	0.0	0.0
S	WS(12)	BG	- 1.10^4	1.10^4
S	WS(13)	BG	- 1.10^2	1.10^2
S	WS(14)	BG	- 10.	10.
S	WS(15)	BG	0.0	0.0
S	BETU(6)	BG	- 4.10^4	4.10^4
S	DSIG(6)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(7)	BG	- 1.10^{-1}	1.10^{-1}
S	DD(8)	BG	0.0	0.0
S	DD(9)	BG	0.0	0.0
S	DD(10)	BG	0.0	0.0
S	DD(11)	BG	0.0	0.0
S	DD(12)	BG	- 1.10^4	1.10^{-4}
S	DD(13)	BG	- 1.10^{-5}	1.10^{-5}
S	DD(14)	BG	- 1.10^{-8}	1.10^{-8}
S	DD(15)	BG	0.0	0.0
S	DD(16)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(17)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(18)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(19)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(20)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(21)	BG	0.0	0.0
S	DD(22)	BG	0.0	0.0
S	DD(23)	BG	0.0	0.0
S	DD(24)	BG	0.0	0.0
S	DD(25)	BG	0.0	0.0
S	DD(26)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	DD(27)	BG	0.0	0.0
S	WS(8)	BG	0.0	0.0
S	WS(9)	BG	0.0	0.0
S	WS(10)	BG	0.0	0.0
S	WS(11)	BG	0.0	0.0
S	WS(12)	BG	-200.	200.
S	WS(13)	BG	- 50.	50.
S	WS(14)	BG	- 1.	1.
S	WS(15)	BG	0.0	0.0
S	BETU(7)	BG	- 1.10^4	1.10^4
S	DSIG(7)	BG	- 1.10^{-1}	1.10^{-1}
S	DD(8)	BG	-100.	100.
S	DD(9)	BG	0.0	0.0
S	DD(10)	BG	0.0	0.0
S	DD(11)	BG	0.0	0.0
S	DD(12)	BG	0.0	0.0
S	DD(13)	BG	0.0	0.0
S	DD(14)	BG	0.0	0.0
S	DD(15)	BG	0.0	0.0
S	DD(16)	BG	0.0	0.0
S	DD(17)	BG	0.0	0.0
S	DD(18)	BG	0.0	0.0
S	DD(19)	BG	0.0	0.0
S	DD(20)	BG	0.0	0.0
S	DD(21)	BG	-100.	100.
S	DD(22)	BG	0.0	0.0
S	DD(23)	BG	0.0	0.0
S	DD(24)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	DD(25)	BG	0.0	0.0
S	DD(26)	BG	0.0	0.0
S	DD(27)	BG	0.0	0.0
S	WS(9)	BG	0.0	0.0
S	WS(10)	BG	0.0	0.0
S	WS(11)	BG	0.0	0.0
S	WS(12)	BG	0.0	0.0
S	WS(13)	BG	0.0	0.0
S	WS(14)	BG	0.0	0.0
S	WS(15)	BG	0.0	0.0
S	BETU(8)	BG	- 1.10 ⁻⁴	1.10 ⁻⁴
S	DSIG(8)	BG	-500.	500.
S	DD(9)	BG	- 1.	1.
S	DD(10)	BG	0.0	0.0
S	DD(11)	BG	0.0	0.0
S	DD(12)	BG	0.0	0.0
S	DD(13)	BG	0.0	0.0
S	DD(14)	BG	0.0	0.0
S	DD(15)	BG	0.0	0.0
S	DD(16)	BG	0.0	0.0
S	DD(17)	BG	0.0	0.0
S	DD(18)	BG	0.0	0.0
S	DD(19)	BG	0.0	0.0
S	DD(20)	BG	0.0	0.0
S	DD(21)	BG	0.0	0.0
S	DD(22)	BG	- 6.10 ⁻²	6.10 ⁻²
S	DD(23)	BG	0.0	0.0
S	DD(24)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	DD(25)	BG	0.0	0.0
S	DD(26)	BG	0.0	0.0
S	DD(27)	BG	0.0	0.0
S	WS(10)	BG	0.0	0.0
S	WS(11)	BG	0.0	0.0
S	WS(12)	BG	0.0	0.0
S	WS(13)	BG	0.0	0.0
S	WS(14)	BG	0.0	0.0
S	WS(15)	BG	0.0	0.0
S	BETU(9)	BG	-600.	600.
S	DSIG(9)	BG	- 1.10^{-1}	1.10^{-1}
S	DD(10)	BG	- 40.	40.
S	DD(11)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(12)	BG	0.0	0.0
S	DD(13)	BG	0.0	0.0
S	DD(14)	BG	0.0	0.0
S	DD(15)	BG	0.0	0.0
S	DD(16)	BG	0.0	0.0
S	DD(17)	BG	0.0	0.0
S	DD(18)	BG	0.0	0.0
S	DD(19)	BG	0.0	0.0
S	DD(20)	BG	0.0	0.0
S	DD(21)	BG	0.0	0.0
S	DD(22)	BG	0.0	0.0
S	DD(23)	BG	- 4.	4.
S	DD(24)	BG	0.0	0.0
S	DD(25)	BG	0.0	0.0
S	DD(26)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	DD(27)	BG	0.0	0.0
S	WS(11)	BG	- 1.10^{-2}	1.10^{-2}
S	WS(12)	BG	0.0	0.0
S	WS(13)	BG	0.0	0.0
S	WS(14)	BG	0.0	0.0
S	WS(15)	BG	0.0	0.0
S	BETU(10)	BG	- 1.10^{-2}	1.10^{-2}
S	DSIG(10)	BG	- 20.	20.
S	DD(11)	BG	- 50.	50.
S	DD(12)	BG	0.0	0.0
S	DD(13)	BG	0.0	0.0
S	DD(14)	BG	0.0	0.0
S	DD(15)	BG	0.0	0.0
S	DD(16)	BG	0.0	0.0
S	DD(17)	BG	0.0	0.0
S	DD(18)	BG	0.0	0.0
S	DD(19)	BG	0.0	0.0
S	DD(20)	BG	0.0	0.0
S	DD(21)	BG	0.0	0.0
S	DD(22)	BG	0.0	0.0
S	DD(23)	BG	- 1.10^{-1}	1.10^{-1}
S	DD(24)	BG	- 4.	4.
S	DD(25)	BG	0.0	0.0
S	DD(26)	BG	0.0	0.0
S	DD(27)	BG	0.0	0.0
S	WS(12)	BG	0.0	0.0
S	WS(13)	BG	0.0	0.0
S	WS(14)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	WS(15)	BG	0.0	0.0
S	BETU(11)	BG	- 1.10 ⁻²	1.10 ⁻²
S	- DSIG(11)	BG	- 20.	20.
S	DD(12)	BG	-500.	500.
S	DD(13)	BG	- 10.	10.
S	DD(14)	BG	- 1.	1.
S	DD(15)	BG	0.0	0.0
S	DD(16)	BG	- 50.	50.
S	DD(17)	BG	- 10.	10.
S	DD(18)	BG	- 1.	1.
S	DD(19)	BG	- 10.	10.
S	DD(20)	BG	- 1.	1.
S	DD(21)	BG	0.0	0.0
S	DD(22)	BG	0.0	0.0
S	DD(23)	BG	0.0	0.0
S	DD(24)	BG	0.0	0.0
S	DD(25)	BG	0.0	0.0
S	DD(26)	BG	0.0	0.0
S	DD(27)	BG	0.0	0.0
S	WS(13)	BG	- 1.	1.
S	WS(14)	BG	- 1.10 ⁻¹	1.10 ⁻¹
S	WS(15)	BG	0.0	0.0
S	BETU(12)	BG	- 1.	1.
S	DSIG(12)	BG	-500.	500.
S	DD(13)	BG	- 10.	10.
S	DD(14)	BG	- 5.	5.
S	DD(15)	BG	0.0	0.0
S	DD(16)	BG	- 1.10 ⁻¹	1.10 ⁻¹

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	DD(17)	BG	- 1.10^{-1}	1.10^{-1}
S	DD(18)	BG	- 1.10^{-1}	1.10^{-1}
S	DD(19)	BG	- 1.10^{-1}	1.10^{-1}
S	DD(20)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(21)	BG	0.0	0.0
S	DD(22)	BG	0.0	0.0
S	DD(23)	BG	0.0	0.0
S	DD(24)	BG	0.0	0.0
S	DD(25)	BG	- 5.	5.
S	DD(26)	BG	0.0	0.0
S	DD(27)	BG	0.0	0.0
S	WS(14)	BG	- .2	.2
S	WS(15)	BG	0.0	0.0
S	BETU(13)	BG	- 1.	1.
S	DSIG(13)	BG	- 10.	10.
S	DD(14)	BG	- 5.	5.
S	DD(15)	BG	0.0	0.0
S	DD(16)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(17)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(18)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(19)	BG	- 1.10^{-2}	1.10^{-2}
S	DD(20)	BG	- 1.10^{-3}	1.10^{-3}
S	DD(21)	BG	0.0	0.0
S	DD(22)	BG	0.0	0.0
S	DD(23)	BG	0.0	0.0
S	DD(24)	BG	0.0	0.0
S	DD(25)	BG	- .01	.01
S	DD(26)	BG	- .002	.002

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	DD(27)	BG	0.0	0.0
S	WS(15)	BG	0.0	0.0
S	BETU(14)	BG	- 5.10 ³	5.10 ³
S	DSIG(14)	BG	- 1.	1.
S	DD(15)	BG	-400.	400.
S	BETU(15)	BG	- 1.10 ⁻³	1.10 ⁻³
S	DSIG(15)	BG	-200.	200.
S	WW(1, 1)	BG	- 1.10 ²	1.10 ²
S	WW(2, 1)	BG	- 50.	50.
S	WW(3, 1)	BG	- 10.	10.
S	WW(4, 1)	BG	- 10.	10.
S	WW(5, 1)	BG	- 1.	1.
S	WW(6, 1)	BG	- 1.	1.
S	WW(7, 1)	BG	- 1.	1.
S	WW(8, 1)	BG	0.0	0.0
S	WW(9, 1)	BG	0.0	0.0
S	WW(10, 1)	BG	0.0	0.0
S	WW(11, 1)	BG	0.0	0.0
S	WW(12, 1)	BG	- 10.	10.
S	WW(13, 1)	BG	- 1.	1.
S	WW(14, 1)	BG	- 1.10 ⁻³	1.10 ⁻³
S	WW(15, 1)	BG	0.0	0.0
S	WW(1, 2)	BG	- 50.	50.
S	WW(2, 2)	BG	-100.	100.
S	WW(3, 2)	BG	- 10.	10.
S	WW(4, 2)	BG	- 10.	10.
S	WW(5, 2)	BG	- 1.	1.
S	WW(6, 2)	BG	- 1.	1.

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	WW(7, 2)	BG	- 1.	1.
S	WW(8, 2)	BG	0.0	0.0
S	WW(9, 2)	BG	0.0	0.0
S	WW(10, 2)	BG	0.0	0.0
S	WW(11, 2)	BG	0.0	0.0
S	WW(12, 2)	BG	- 10.	10.
S	WW(13, 2)	BG	- 1.	1.
S	WW(14, 2)	BG	- 1.10^{-3}	1.10^{-3}
S	WW(15, 2)	BG	0.0	0.0
S	WW(1, 3)	BG	- 5.	5.
S	WW(2, 3)	BG	- 5.	5.
S	WW(3, 3)	BG	- 5.	5.
S	WW(4, 3)	BG	- 1.	1.
S	WW(5, 3)	BG	- 1.	1.
S	WW(6, 3)	BG	- 1.	1.
S	WW(7, 3)	BG	- 1.	1.
S	WW(8, 3)	BG	0.0	0.0
S	WW(9, 3)	BG	0.0	0.0
S	WW(10, 3)	BG	0.0	0.0
S	WW(11, 3)	BG	0.0	0.0
S	WW(12, 3)	BG	- 1.	1.
S	WW(13, 3)	BG	- .1	.1
S	WW(14, 3)	BG	- 1.10^{-3}	1.10^{-3}
S	WW(15, 3)	BG	0.0	0.0
S	WW(1, 4)	BG	- 5.	5.
S	WW(2, 4)	BG	- 5.	5.
S	WW(3, 4)	BG	- 5.	5.
S	WW(4, 4)	BG	- 5.	5.

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	WW(5, 4)	BG	- 1.	1.
S	WW(6, 4)	BG	- 1.	1.
S	WW(7, 4)	BG	- 1.	1.
S	WW(8, 4)	BG	0.0	0.0
S	WW(9, 4)	BG	0.0	0.0
S	WW(10, 4)	BG	0.0	0.0
S	WW(11, 4)	BG	0.0	0.0
S	WW(12, 4)	BG	- 1.	1.
S	WW(13, 4)	BG	- 1.	1.
S	WW(14, 4)	BG	- 1.10^{-3}	1.10^{-3}
S	WW(15, 4)	BG	0.0	0.0
S	WW(1, 5)	BG	- 1.10^{-2}	1.10^{-2}
S	WW(2, 5)	BG	- 1.10^{-2}	1.10^{-2}
S	WW(3, 5)	BG	- 1.10^{-2}	1.10^{-2}
S	WW(4, 5)	BG	- 1.10^{-2}	1.10^{-2}
S	WW(5, 5)	BG	- 1.10^{-2}	1.10^{-2}
S	WW(6, 5)	BG	- 1.10^{-3}	1.10^{-3}
S	WW(7, 5)	BG	- 1.10^{-3}	1.10^{-3}
S	WW(8, 5)	BG	0.0	0.0
S	WW(9, 5)	BG	0.0	0.0
S	WW(10, 5)	BG	0.0	0.0
S	WW(11, 5)	BG	0.0	0.0
S	WW(12, 5)	BG	- 1.10^{-3}	1.10^{-3}
S	WW(13, 5)	BG	- 1.10^{-3}	1.10^{-3}
S	WW(14, 5)	BG	- 1.10^{-6}	1.10^{-6}
S	WW(15, 5)	BG	0.0	0.0
S	WW(1, 6)	BG	- 1.10^{-2}	1.10^{-2}
S	WW(2, 6)	BG	- 1.10^{-2}	1.10^{-2}

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	WW(3, 6)	BG	- 1.10 ⁻²	1.10 ⁻²
S	WW(4, 6)	BG	- 1.10 ⁻²	1.10 ⁻²
S	WW(5, 6)	BG	- 1.10 ⁻³	1.10 ⁻³
S	WW(6, 6)	BG	- 1.10 ⁻²	1.10 ⁻²
S	WW(7, 6)	BG	- 1.10 ⁻³	1.10 ⁻³
S	WW(8, 6)	BG	0.0	0.0
S	WW(9, 6)	BG	0.0	0.0
S	WW(10, 6)	BG	0.0	0.0
S	WW(11, 6)	BG	0.0	0.0
S	WW(12, 6)	BG	- 1.10 ⁻³	1.10 ⁻³
S	WW(13, 6)	BG	- 1.10 ⁻³	1.10 ⁻³
S	WW(14, 6)	BG	- 1.10 ⁻⁶	1.10 ⁻⁶
S	WW(15, 6)	BG	0.0	0.0
S	WW(1, 7)	BG	- 1.10 ⁻²	1.10 ⁻²
S	WW(2, 7)	BG	- 1.10 ⁻²	1.10 ⁻²
S	WW(3, 7)	BG	- 1.10 ⁻²	1.10 ⁻²
S	WW(4, 7)	BG	- 1.10 ⁻²	1.10 ⁻²
S	WW(5, 7)	BG	- 1.10 ⁻³	1.10 ⁻³
S	WW(6, 7)	BG	- 1.10 ⁻³	1.10 ⁻³
S	WW(7, 7)	BG	- 1.10 ⁻²	1.10 ⁻²
S	WW(8, 7)	BG	0.0	0.0
S	WW(9, 7)	BG	0.0	0.0
S	WW(10, 7)	BG	0.0	0.0
S	WW(11, 7)	BG	0.0	0.0
S	WW(12, 7)	BG	- 1.10 ⁻³	1.10 ⁻³
S	WW(13, 7)	BG	- 1.10 ⁻³	1.10 ⁻³
S	WW(14, 7)	BG	- 1.10 ⁻⁶	1.10 ⁻⁶
S	WW(15, 7)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	WW(1, 8)	BG	0.0	0.0
S	WW(2, 8)	BG	0.0	0.0
S	WW(3, 8)	BG	0.0	0.0
S	WW(4, 8)	BG	0.0	0.0
S	WW(5, 8)	BG	0.0	0.0
S	WW(6, 8)	BG	0.0	0.0
S	WW(7, 8)	BG	0.0	0.0
S	WW(8, 8)	BG	-500.	500.
S	WW(9, 8)	BG	0.0	0.0
S	WW(10, 8)	BG	0.0	0.0
S	WW(11, 8)	BG	0.0	0.0
S	WW(12, 8)	BG	0.0	0.0
S	WW(13, 8)	BG	0.0	0.0
S	WW(14, 8)	BG	0.0	0.0
S	WW(15, 8)	BG	0.0	0.0
S	WW(1, 9)	BG	0.0	0.0
S	WW(2, 9)	BG	0.0	0.0
S	WW(3, 9)	BG	0.0	0.0
S	WW(4, 9)	BG	0.0	0.0
S	WW(5, 9)	BG	0.0	0.0
S	WW(6, 9)	BG	0.0	0.0
S	WW(7, 9)	BG	0.0	0.0
S	WW(8, 9)	BG	0.0	0.0
S	WW(9, 9)	BG	.1	.1
S	WW(10, 9)	BG	0.0	0.0
S	WW(11, 9)	BG	0.0	0.0
S	WW(12, 9)	BG	0.0	0.0
S	WW(13, 9)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	WW(14, 9)	BG	0.0	0.0
S	WW(15, 9)	BG	0.0	0.0
S	WW(1, 10)	BG	0.0	0.0
S	WW(2, 10)	BG	0.0	0.0
S	WW(3, 10)	BG	0.0	0.0
S	WW(4, 10)	BG	0.0	0.0
S	WW(5, 10)	BG	0.0	0.0
S	WW(6, 10)	BG	0.0	0.0
S	WW(7, 10)	BG	0.0	0.0
S	WW(8, 10)	BG	0.0	0.0
S	WW(9, 10)	BG	0.0	0.0
S	WW(10, 10)	BG	- 20.	20.
S	WW(11, 10)	BG	- 1.	1.
S	WW(12, 10)	BG	0.0	0.0
S	WW(13, 10)	BG	0.0	0.0
S	WW(14, 10)	BG	0.0	0.0
S	WW(15, 10)	BG	0.0	0.0
S	WW(1, 11)	BG	0.0	0.0
S	WW(2, 11)	BG	0.0	0.0
S	WW(3, 11)	BG	0.0	0.0
S	WW(4, 11)	BG	0.0	0.0
S	WW(5, 11)	BG	0.0	0.0
S	WW(6, 11)	BG	0.0	0.0
S	WW(7, 11)	BG	0.0	0.0
S	WW(8, 11)	BG	0.0	0.0
S	WW(9, 11)	BG	0.0	0.0
S	WW(10, 11)	BG	- 1.	1.
S	WW(11, 11)	BG	- 20.	20.

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	WW(12, 11)	BG	0.0	0.0
S	WW(13, 11)	BG	0.0	0.0
S	WW(14, 11)	BG	0.0	0.0
S	WW(15, 11)	BG	0.0	0.0
S	WW(1, 12)	BG	-100.	100.
S	WW(2, 12)	BG	-100.	100.
S	WW(3, 12)	BG	- 50.	50.
S	WW(4, 12)	BG	- 50.	50.
S	WW(5, 12)	BG	- 1.	1.
S	WW(6, 12)	BG	- 1.	1.
S	WW(7, 12)	BG	- 1.	1.
S	WW(8, 12)	BG	0.0	0.0
S	WW(9, 12)	BG	0.0	0.0
S	WW(10, 12)	BG	0.0	0.0
S	WW(11, 12)	BG	0.0	0.0
S	WW(12, 12)	BG	-500.	500.
S	WW(13, 12)	BG	- 50.	50.
S	WW(14, 12)	BG	- 1.	1.
S	WW(15, 12)	BG	0.0	0.0
S	WW(1, 13)	BG	- 50.	50.
S	WW(2, 13)	BG	- 50.	50.
S	WW(3, 13)	BG	- 50.	50.
S	WW(4, 13)	BG	- 50.	50.
S	WW(5, 13)	BG	- 1.	1.
S	WW(6, 13)	BG	- 1.	1.
S	WW(7, 13)	BG	- 1.	1.
S	WW(8, 13)	BG	0.0	0.0
S	WW(9, 13)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	WW(10, 13)	BG	0.0	0.0
S	WW(11, 13)	BG	0.0	0.0
S	WW(12, 13)	BG	- 50.	50.
S	WW(13, 13)	BG	- 10.	10.
S	WW(14, 13)	BG	- 5.	5.
S	WW(15, 13)	BG	0.0	0.0
S	WW(1, 14)	BG	- 1.	1.
S	WW(2, 14)	BG	- 1.	1.
S	WW(3, 14)	BG	- 1.	1.
S	WW(4, 14)	BG	- 1.	1.
S	WW(5, 14)	BG	- 1.10^{-2}	1.10^{-2}
S	WW(6, 14)	BG	- 1.10^{-2}	1.10^{-2}
S	WW(7, 14)	BG	- 1.10^{-2}	1.10^{-2}
S	WW(8, 14)	BG	0.0	0.0
S	WW(9, 14)	BG	0.0	0.0
S	WW(10, 14)	BG	0.0	0.0
S	WW(11, 14)	BG	0.0	0.0
S	WW(12, 14)	BG	- 1.	1.
S	WW(13, 14)	BG	- 1.	1.
S	WW(14, 14)	BG	- 1.	1.
S	WW(15, 14)	BG	0.0	0.0
S	WW(1, 15)	BG	0.0	0.0
S	WW(2, 15)	BG	0.0	0.0
S	WW(3, 15)	BG	0.0	0.0
S	WW(4, 15)	BG	0.0	0.0
S	WW(5, 15)	BG	0.0	0.0
S	WW(6, 15)	BG	0.0	0.0
S	WW(7, 15)	BG	0.0	0.0

TABLE 3 (cont.)

BACKGROUND

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Min</u>	<u>Max</u>
S	WW(8, 15)	BG	0.0	0.0
S	WW(9, 15)	BG	0.0	0.0
S	WW(10, 15)	BG	0.0	0.0
S	WW(11, 15)	BG	0.0	0.0
S	WW(12, 15)	BG	0.0	0.0
S	WW(13, 15)	BG	0.0	0.0
S	WW(14, 15)	BG	0.0	0.0
S	WW(15, 15)	BG	-200.	200.

The following storage cells are required for computations in the background loop although the values are generated in other sections:

<u>Type</u>	<u>Name</u>	<u>Origin</u>	<u>Number of Cells</u>
I	IND(J)	MG	J = 1, 4
V	YR(J)	MG	J = 1, 15
V	H1(J)	MG	J = 1, 15
V	H2(J)	MG	J = 1, 15
V	H3(J)	MG	J = 1, 15
V	H4(J)	MG	J = 1, 15
V	APS(J)	MG	J = 1, 15
P	NE(J)	INITIAL	J = 1, 27
P	NR(J)	INITIAL	J = 1, 15
P	B(J)	INITIAL	J = 1, 12

The total number of storage cells required for the background computations are given below by type:

Variables	341
Scratch Variables	708
Permanent	54
Integer	4

VIII. COMPARISON OF KALMAN FILTER NAVIGATION USING 16 BITS VERSUS 60 BITS

In this section, we illustrate the performance of the Kalman filter in the role of navigation utilizing an onboard computer of the GE-701 type with 16-bit word length, and compare the results with the same navigation method utilizing a full 60-bit word length.

The results of the simulation are shown on a CALCOMP plot for a typical B-737 landing covering the phases of a bank-turn approach, glide slope capture, descent, flare, decrab, rollout and turnoff. The variables plotted versus time are:

- 1) Glide slope deviation, meters
- 2) Aircraft pitch history, degrees
- 3) Elevator control, degrees
- 4) Vertical descent rate, m/sec
- 5) Ground speed, m/sec
- 6) Vertical acceleration, m/sec²
- 7) Cross track deviation, meters
- 8) Aircraft roll, degrees
- 9) Aircraft yaw, degrees
- 10) Rudder and/or nose wheel steering, degrees
- 11) Aileron deflection, degrees
- 12) Altitude, meters

Figures Va and Vb are for the 16-bit simulation. Figures VIa and VIb are for the 60-bit simulation.

It is evident that no serious deterioration of control occurs using the 16-bit computer following the computer logic in this report.

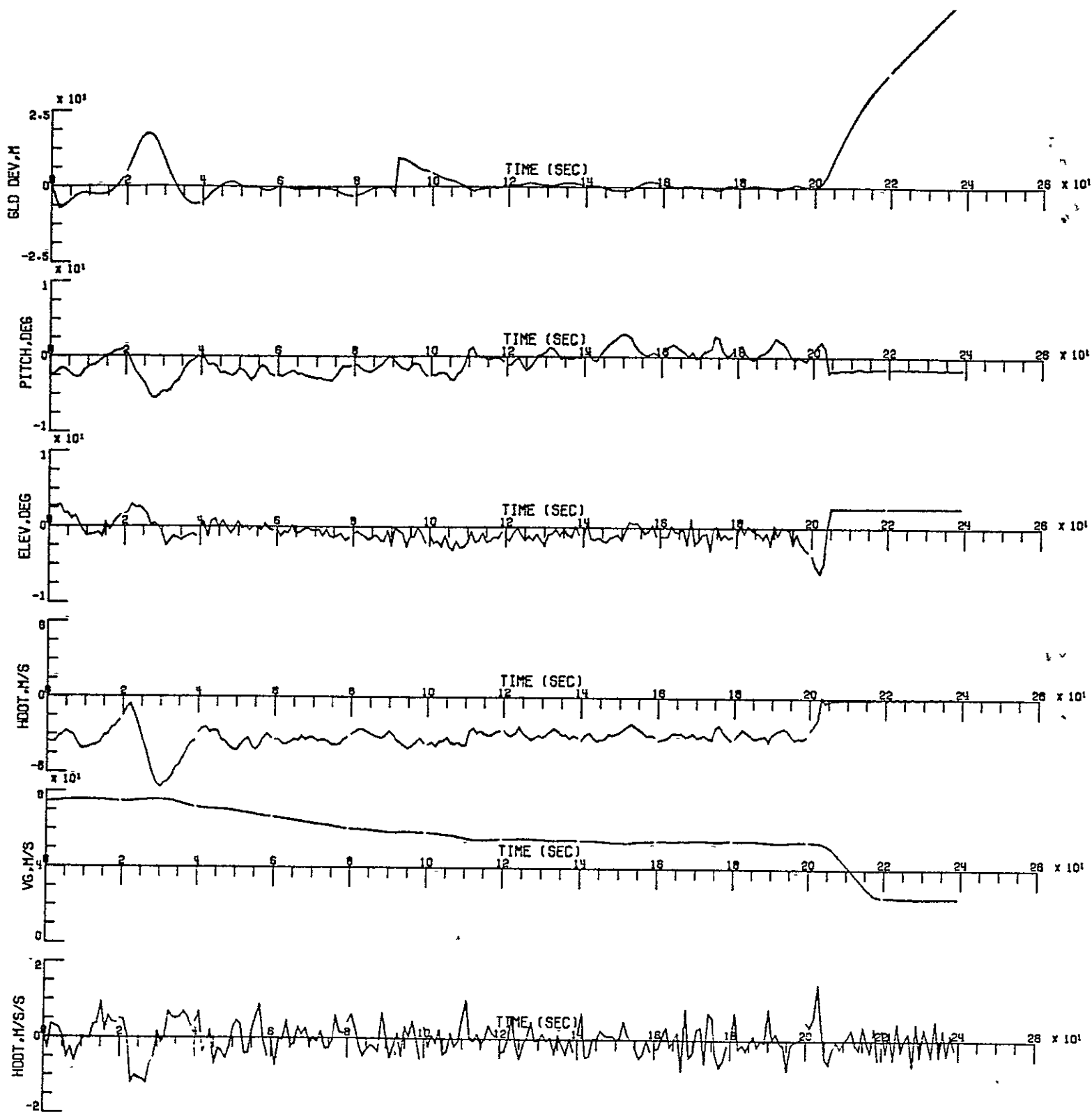


Figure Va. 16-Bit Simulation of the B-737 Kalman Filter

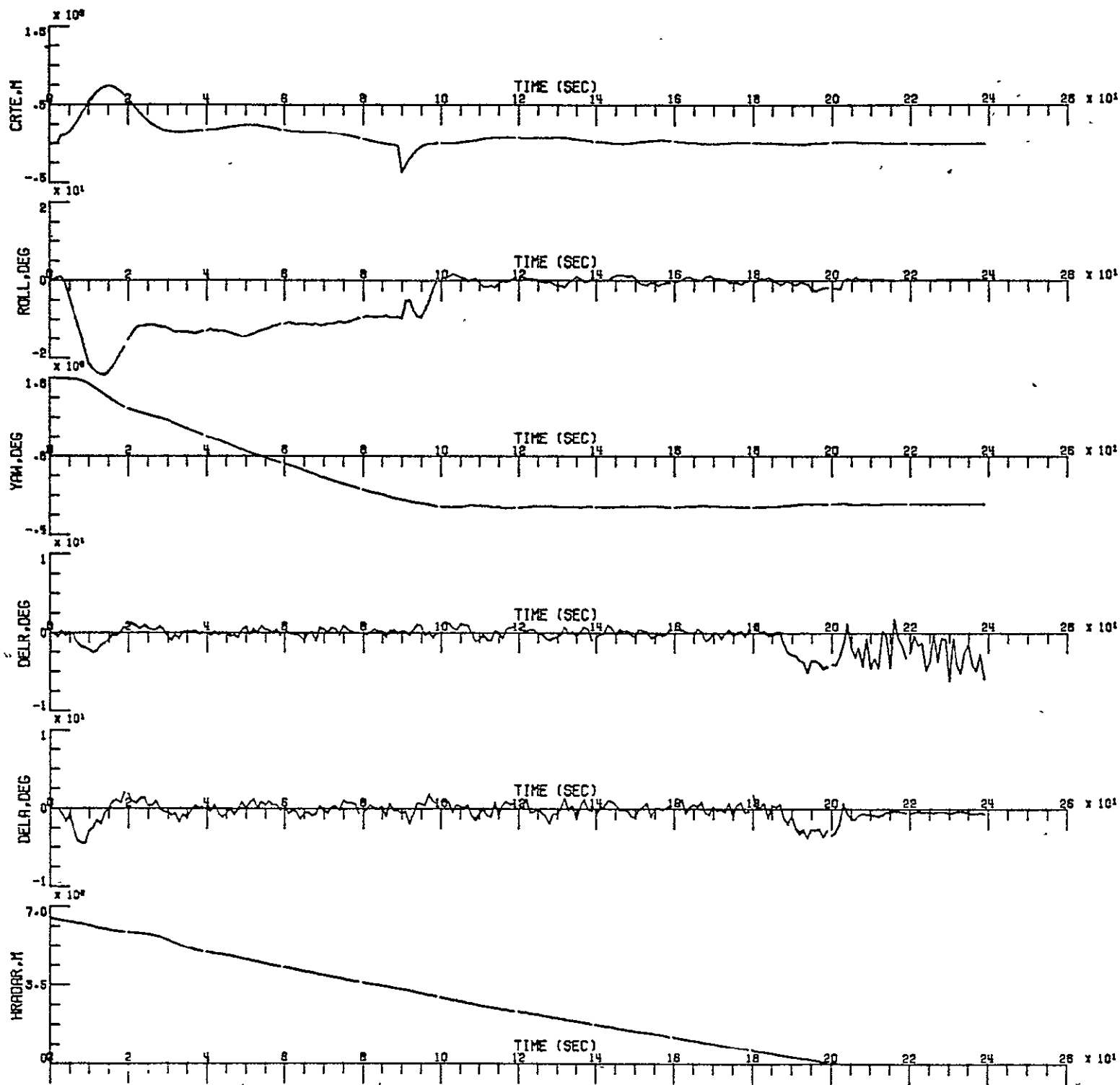


Figure Vb. 16-Bit Simulation of the B-737 Kalman Filter

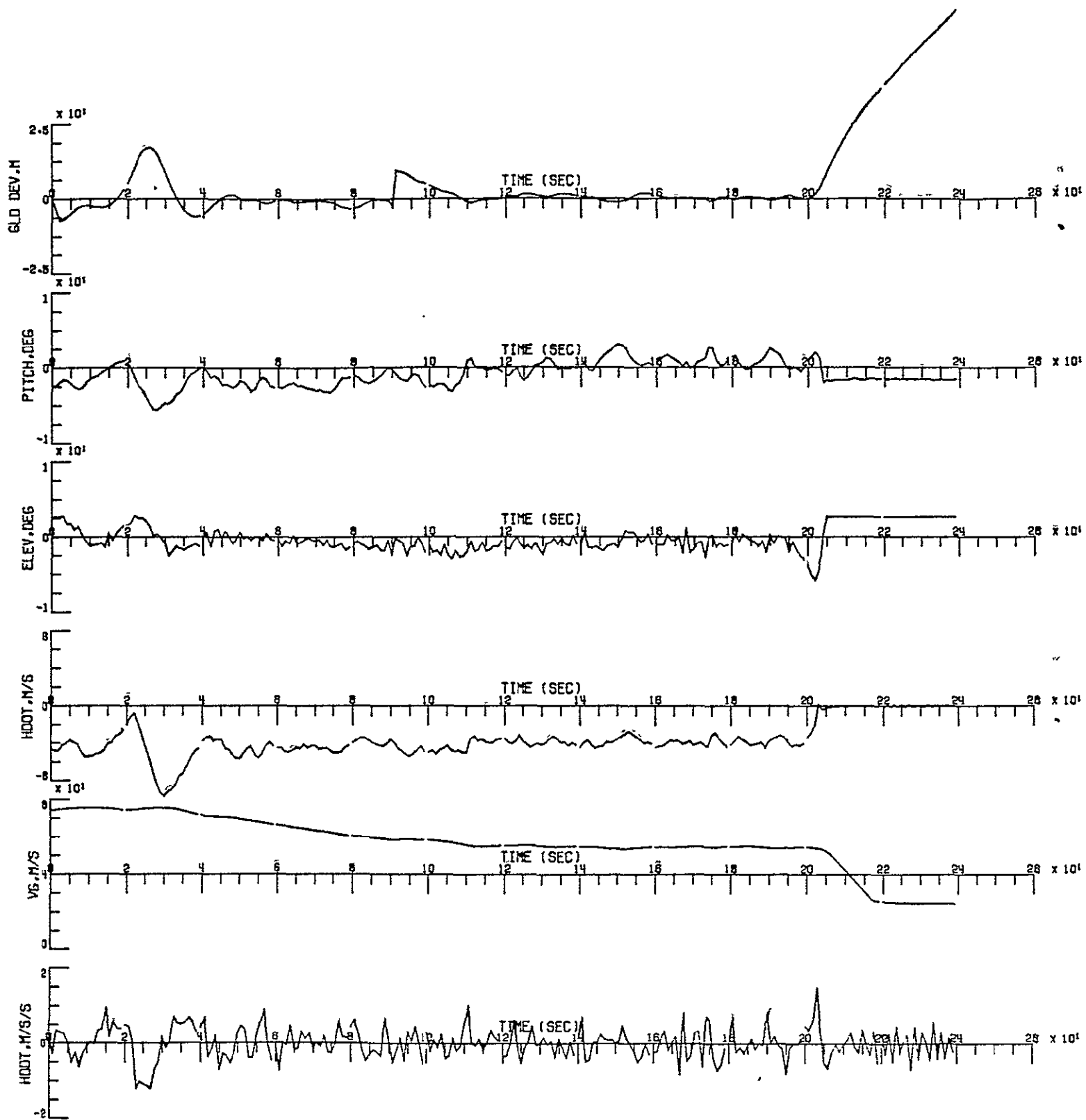


Figure VIa. 60-Bit Simulation of the B-737 Kalman Filter

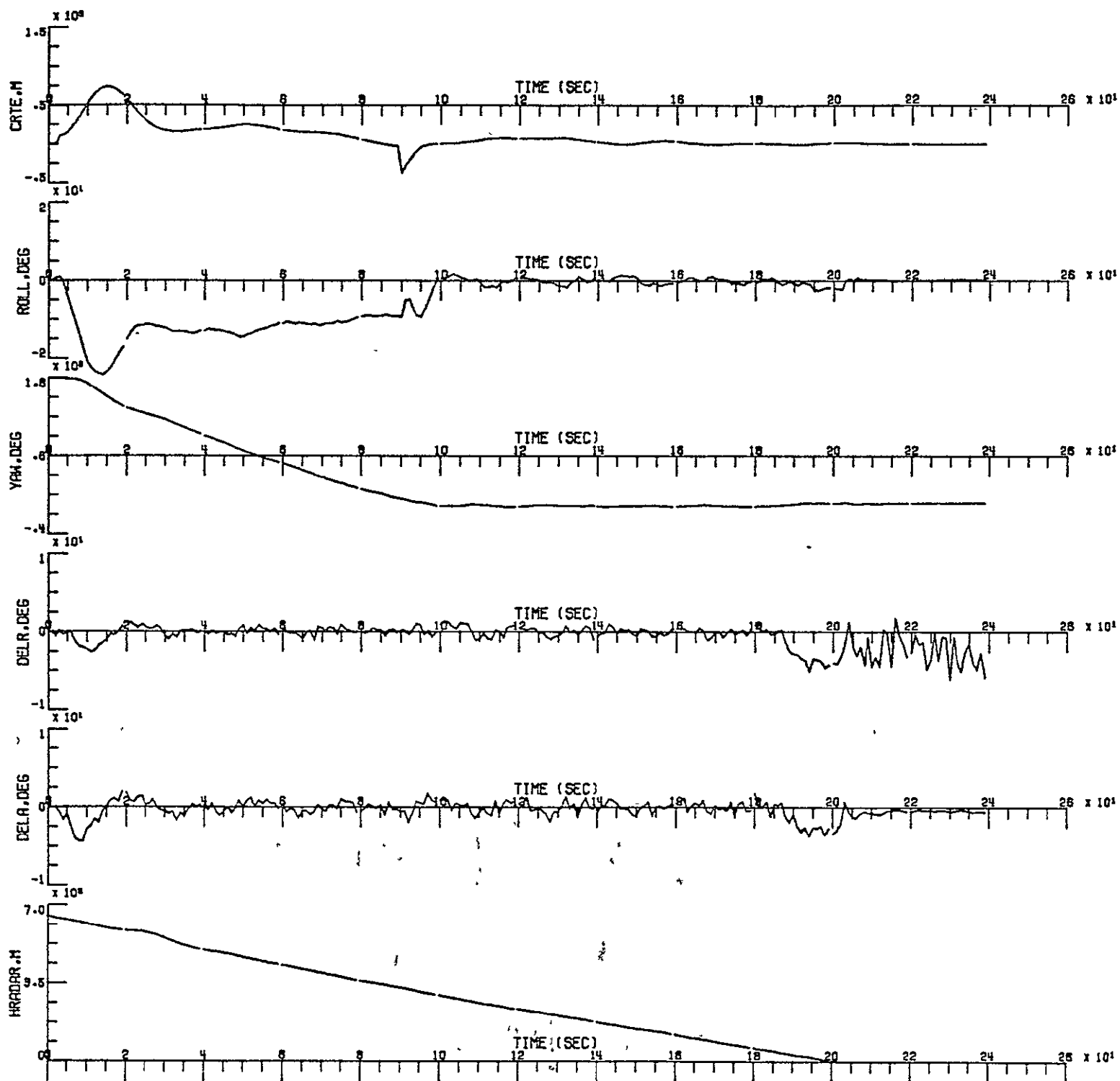


Figure VIb, 60-Bit Simulation of the B-737 Kalman Filter

IX. REFERENCES

1. Pines, S. Schmidt, S. F., and Mann, F.; "Automated Landing, Rollout, and Turnoff Using MLS and Magnetic Cable Sensors," NASA CR-2907, Oct. 1977.
2. Anon.; "MCP-701 Programmer's Reference Manual," General Electric Company, Aircraft Equipment Division, Binghamton, N.Y.

END DATE
FILMED

Jul 11 1978